

《数据库系统》课程设计

--房屋租售管理系统设计与开发

班级： 软件工程1901

学号： 1931053796

姓名： 薛文朝

2021年 6月25日

数据库系统课程设计评阅

|  |  |  |
| --- | --- | --- |
| 一、论文（50分） | | 实际得分 |
| 1.需求分析：给出完整的数据需求、处理需求及安全性和完整性需求，分析详细、正确，数据流程图正确。 | 满 分  （20分） |  |
| 2.概念结构设计：有详细的分析步骤和理论依据，E-R图正确、完备。 | 满 分  （10分） |  |
| 3.逻辑结构设计: 有详细的转换步骤和理论依据，关系模型正确，并确定主码、外码。 | 满 分  （10分） |  |
| 4.物理结构设计:有索引、聚簇、日志、备份等的存储安排和存储结构，内容正确。 | 满 分  （10分） |  |
| 二、PowerDesigner设计图（20分） | | |
| 1.熟练掌握PD绘图工具的使用，有完整的概念模型图和物理模型图，正确、完备。 | 满 分  （20分） |  |
| 三、系统及数据库（20分） | |  |
| 系统运行正常，功能完善，代码正确，根据数据安全性和完整性需要，创建视图、自定义函数、存储过程，设计对象完备正确。 | 满 分  （20分） |  |
| 四、总体评价（10分） | | |
| 课程报告的规范性；总结收获和体会；已解决和尚未解决的问题分析；完善的设想与建议。 | 满 分  （10分） |  |
| 总分 | |  |
| 阅卷教师签名 | |  |

目录

[一、系统需求分析 1](#_Toc487448818)

[（一）需求概述 1](#_Toc487448819)

[（二）业务流分析 1](#_Toc487448824)

[（三）数据流分析 3](#_Toc487448831)

[（四）数据字典 4](#_Toc487448833)

[二、数据库概念结构设计 5](#_Toc487448961)

[（一）实体分析 5](#_Toc487448962)

[（二）属性分析 5](#_Toc487448963)

[（三）联系分析 7](#_Toc487448964)

[（四）概念模型分析（.CDM图） 8](#_Toc487448965)

[三、数据库逻辑结构设计 8](#_Toc487448968)

[（一）概念模型转化为逻辑模型 8](#_Toc487448969)

[1. 一对一关系的转化 8](#_Toc487448970)

[2. 一对多关系的转化 8](#_Toc487448971)

[3. 多对多关系的转化 9](#_Toc487448972)

[（二）逻辑模型设计（.LDM图） 9](#_Toc487448973)

[四、数据库物理实现 9](#_Toc487448975)

[（一）表设计 9](#_Toc487448976)

[（二）创建表和完整性约束代码设计 11](#_Toc487449098)

[（三）创建视图、索引、存储过程和触发器 13](#_Toc487449105)

[1. 创建视图 13](#_Toc487449106)

[2. 创建索引 14](#_Toc487449109)

[3. 创建存储过程 14](#_Toc487449111)

[五、数据库功能调试 14](#_Toc487449114)

[（一）职工管理模块 14](#_Toc487449115)

[1. 查询职工信息 15](#_Toc487449116)

[2. 查询职工所在单位信息 15](#_Toc487449117)

[3. 查询职工所属工程信息 15](#_Toc487449118)

[（二）工程负责人管理模块 15](#_Toc487449119)

[1. 查询工程信息 16](#_Toc487449120)

[2. 查询工程所需设备信息 16](#_Toc487449122)

[3. 查询工程供应商信息 16](#_Toc487449124)

[4. 查询参与工程的职工信息 16](#_Toc487449126)

[（三）系统管理员管理模块 17](#_Toc487449128)

[1. 对单位的信息管理 17](#_Toc487449129)

[2. 对职工的信息管理 18](#_Toc487449130)

[3. 对供应商的信息管理 19](#_Toc487449131)

[4. 对工程的信息管理 19](#_Toc487449132)

[5. 对设备的信息管理 20](#_Toc487449133)

[6. 对供应关系的信息管理 21](#_Toc487449134)

[六、设计总结 22](#_Toc487449135)

要求一级目录不变，二三级目录自定

# 

# 系统需求分析

1. 需求概述
2. 系统设计理念：

在计算机高速发展的现代化信息化的今天，传统的面对面房屋交易以及显得过时，大量的房源，客户，费用等数据显得复杂且细致，在这庞大的数据量下，手工操作较为繁琐易错。为了推动房屋租售管理的规范化，和信息化，用计算机来对这些数据进行科学、系统的管理势在必行。计算机进行管理，有着高保密性，查找方便，存储量大，寿命长等优秀特点。故开发房屋租售管理系统可以使房屋出售信息得到科学、规范且高效的数字化管理。

1. 系统结构简述：

本数据库主要包含以下的主要功能实体：租客基本信息，房主基本信息。

实体之间的关系转换有：住房-租客基本信息。完成数据的订单的存取的实体有：仓库管理员。

1. 开发工具：

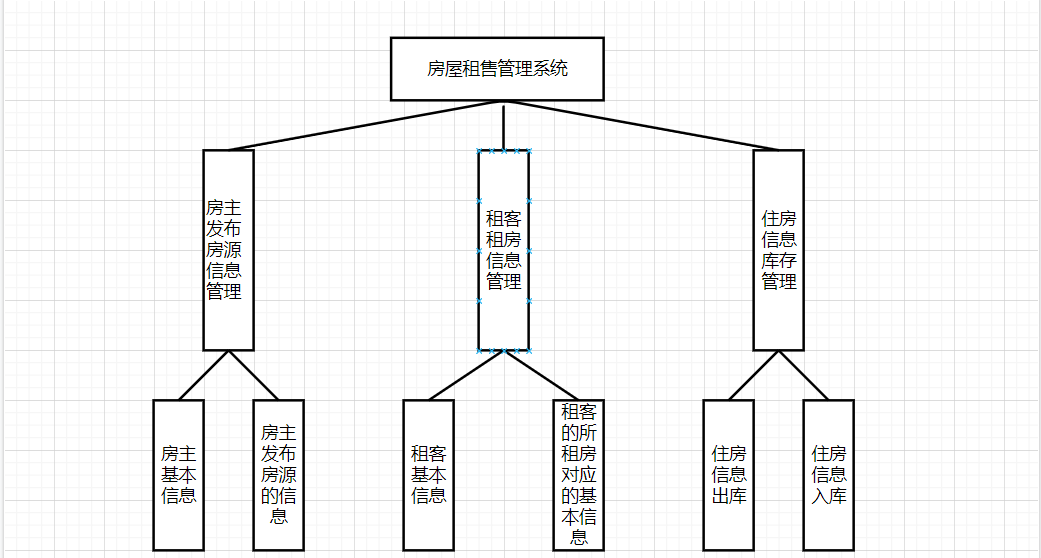
IDEA、MySQL、PowerDesigner、Navicat Premium

系统环境：

Windows、Linux

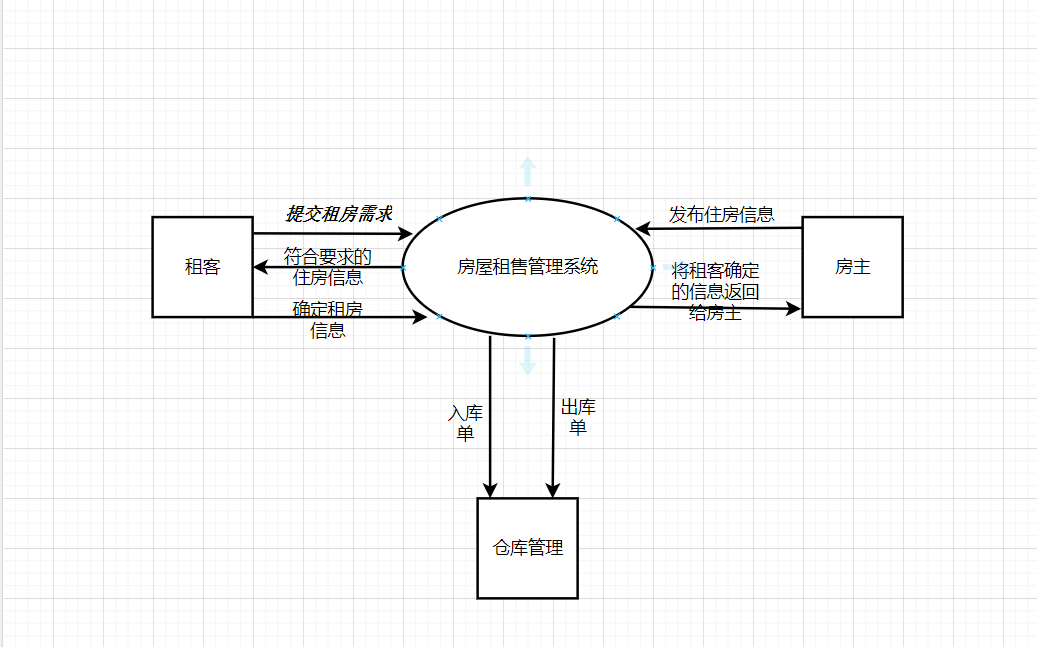
1. 业务流分析

针对于房屋租售管理系统，分别要对住房剩余库存，住房信息，房主信息，租客信息进行查询与处理。

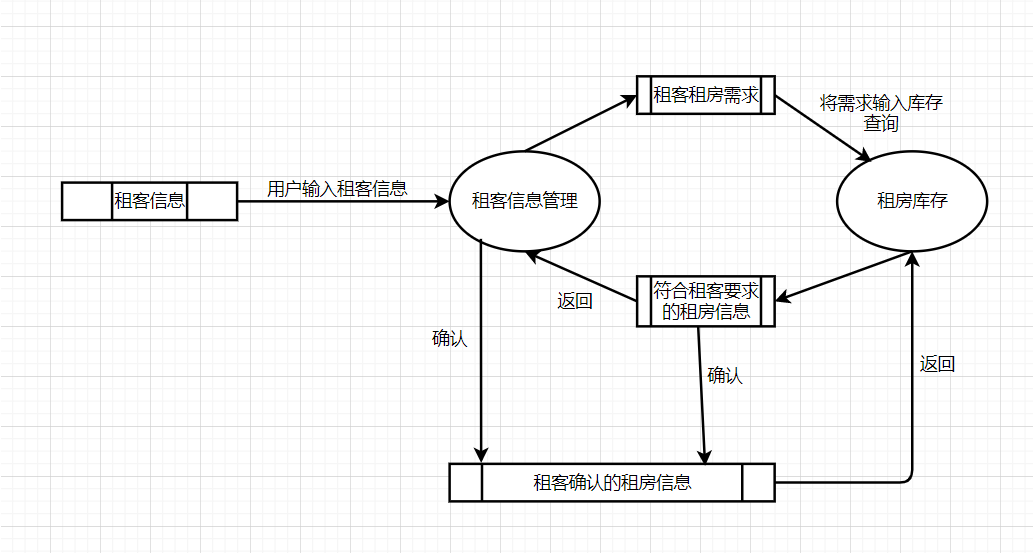


1. 数据流分析

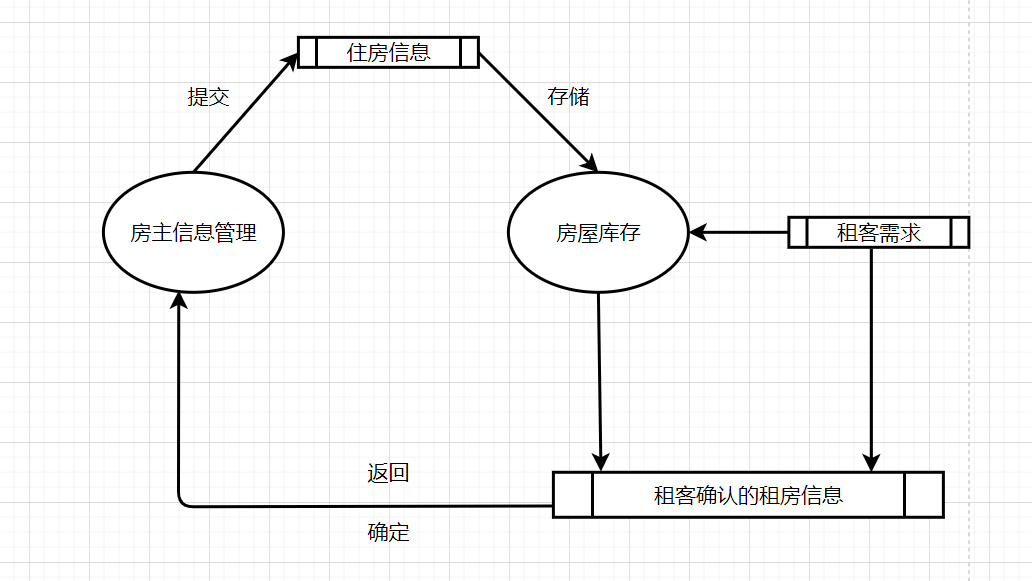
**1.顶层数据流图：**



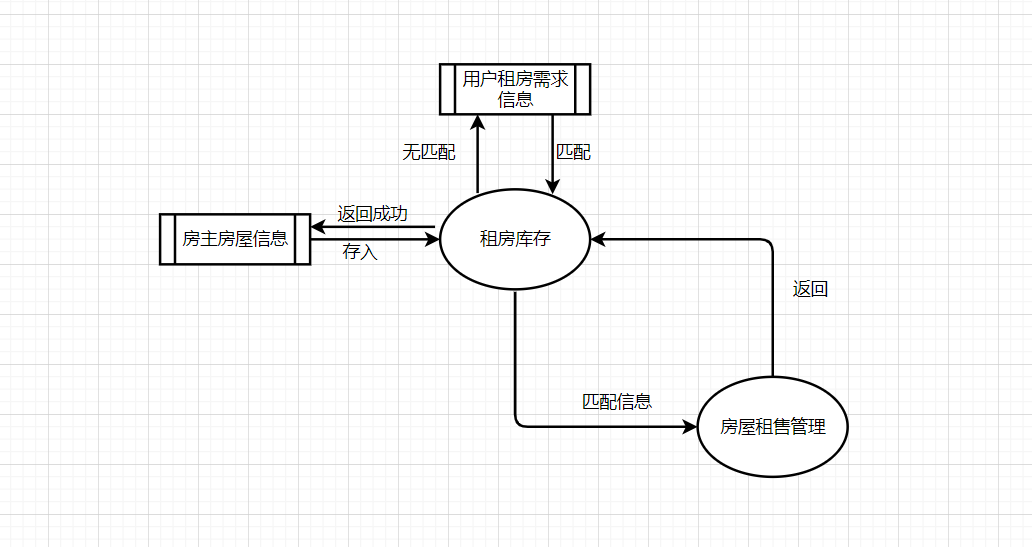
**2.租客层数据流图：**

****

**3.房主层数据流：**

****

**4.仓库管理数据流：**

****

1. 数据字典
   * + 1. 数据项：

关系模式（带有下划线的为主键，所有表中数据均为varchar类型）：

管理员（用户名、密码）

房屋信息（房屋编号、所属区域、房屋名称、房屋类型、房屋面积、装修情况、房屋设施、房屋用途、月租价格、房间状态、备注说明）

用户信息（身份证号、用户姓名、性别、电话）

租赁信息（房屋编号、租户身份证号、租赁开始时间、租赁月数 、月租金）

水电费信息（受理单号、房屋编号、上次电表数、本次电表数、上次水表数、本次水表数、收费金额）

财务收支报（编号、类型、金额、日期）

杂项设置（编号、水费单价、电费单价、宽带费单价、物业费单价、财务余额）

* + - 1. 数据结构：

管理员表（manageruser）：

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 约束 |
| user | 管理员用户名 | varchar（20） | 主键 |
| password | 密码 | varchar（20） | 不为空 |

房屋信息表（roominf）：

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 约束 |
| num | 房屋编号 | varchar（10） | 主键 |
| weizhi | 房屋位置 | varchar（30） | 不为空 |
| mingzi | 房屋名称 | varchar（20） | 不为空 |
| zhonglei | 房屋类型 | varchar（10） | 不为空 |
| mianji | 房屋面积 | varchar（11） | 不为空 |
| zhuangxiu | 装修状况 | varchar（4） | 不为空 |
| sheshi | 屋内设施 | varchar（20） | 不为空 |
| yongtu | 房屋用途 | varchar（4） | 不为空 |
| jiage | 房屋价格 | varchar（11） | 不为空 |
| zhuangtai | 房屋状态 | varchar（4） | 不为空 |
| beizhu | 备注信息 | varchar（40） | 不为空 |

用户信息表（userinf）：

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 约束 |
| sfzid | 身份证号 | varchar（18） | 主键 |
| username | 姓名 | varchar（10） | 不为空 |
| xingbie | 性别 | varchar（2） | 不为空 |
| dianhua | 电话 | varchar（14） | 不为空 |

房屋租赁表（roomrent）：

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 约束 |
| roomnum | 房屋编号 | varchar（10） | 主键 |
| sfznum | 身份证号 | varchar（18） | 主键 |
| starttime | 开始租住时间 | varchar（12） | 不为空 |
| monthnum | 租住月份数 | varchar（4） | 不为空 |
| monthlyrent | 月租金 | varchar（20） | 不为空 |

水电费表（billsinf）：

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 约束 |
| danhao | 受理单号 | varchar（15） | 主键 |
| roomnum | 房屋编号 | varchar（10） | 不为空 |
| predianbiao | 上次电表数 | varchar（15） | 不为空 |
| dianbiao | 本次电表数 | varchar（15） | 不为空 |
| preshuibiao | 上次水表数 | varchar（15） | 不为空 |
| shuibiao | 本次水表数 | varchar（15） | 不为空 |
| jine | 金额 | varchar（20） | 不为空 |

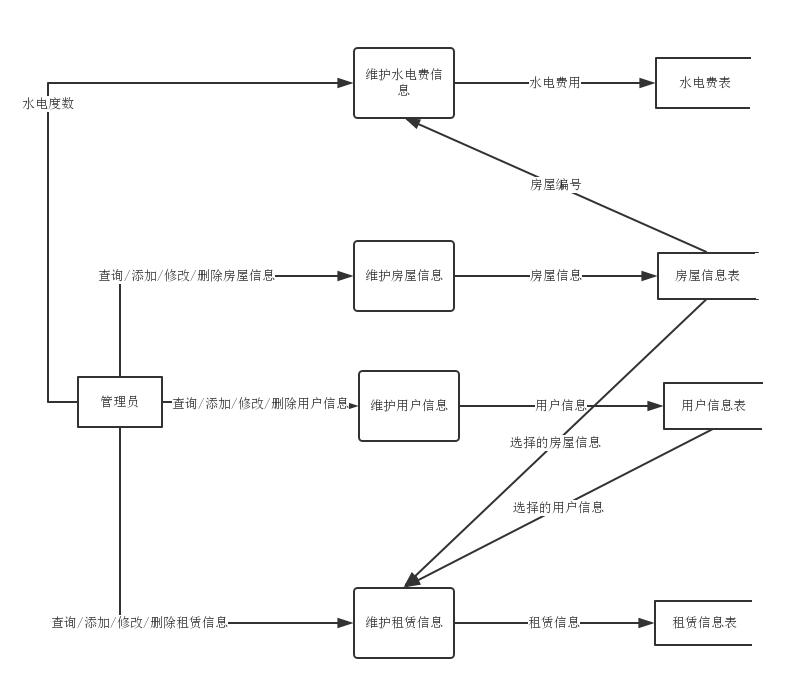
财务报表(caiwu):

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 约束 |
| bianhao | 编号 | varchar（20） | 主键 |
| leixing | 类型 | varchar（2） | 不为空 |
| jine | 金额 | varchar（15） | 不为空 |
| date | 日期 | varchar（12） | 不为空 |

杂项设置表（othersetting）：

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 约束 |
| bianhao | 编号 | varchar（10） | 主键 |
| shuiprice | 水价 | varchar（10） | 不为空 |
| dianprice | 电价 | varchar（10） | 不为空 |
| wangprice | 网费 | varchar（10） | 不为空 |
| wuyeprice | 物业费 | varchar（10） | 不为空 |
| yue | 余额 | varchar（10） | 不为空 |

* + - 1. 数据流：



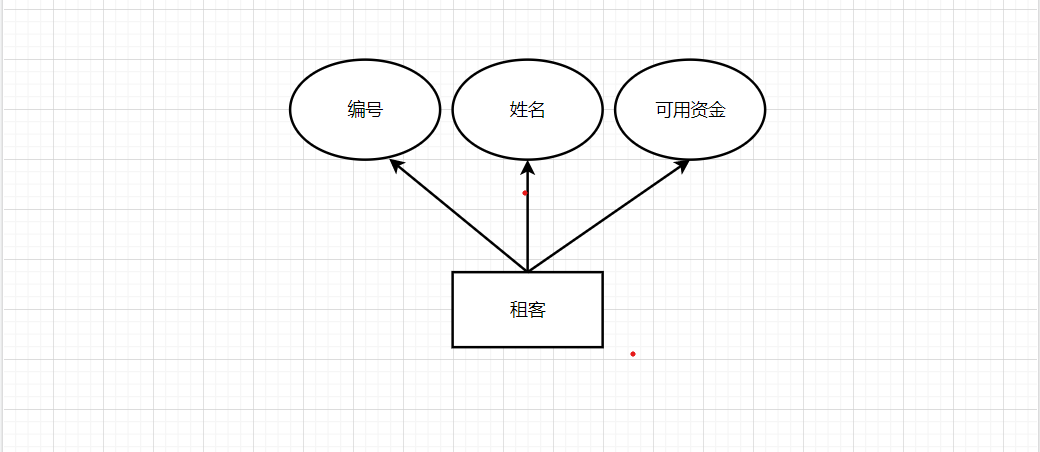
* + - 1. 数据存储：

采用Mysql数据库存储，多为varchar类型数据

# 数据库概念结构设计

* 1. ****实体分析****
     + 1. ****租客实体（tenent表）****

**ER图：**

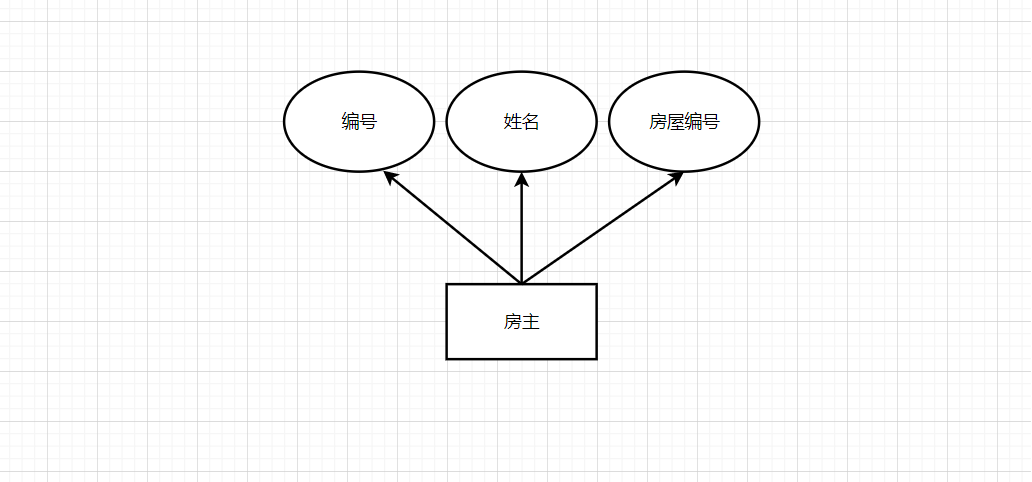
****

**分析：**

**租客实体有着三个重要的属性列分别为：编号（主键约束、自增约束）、姓名（租客的姓名，唯一约束）、可用资金（用户可以调用的资金）。**

* + - 1. **房主实体（owner表）**

**ER图：**

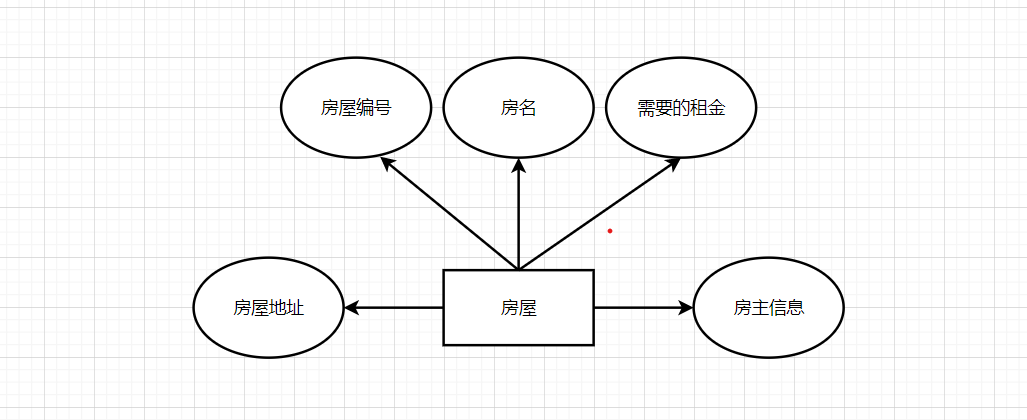
****

**分析：**

**房主实体有四个属性列：编号（主键）、姓名、房屋编号（外键，对应于房屋信息表的编号）、联系电话**

* + - 1. **房屋信息实体（house表）**

**ER图：**

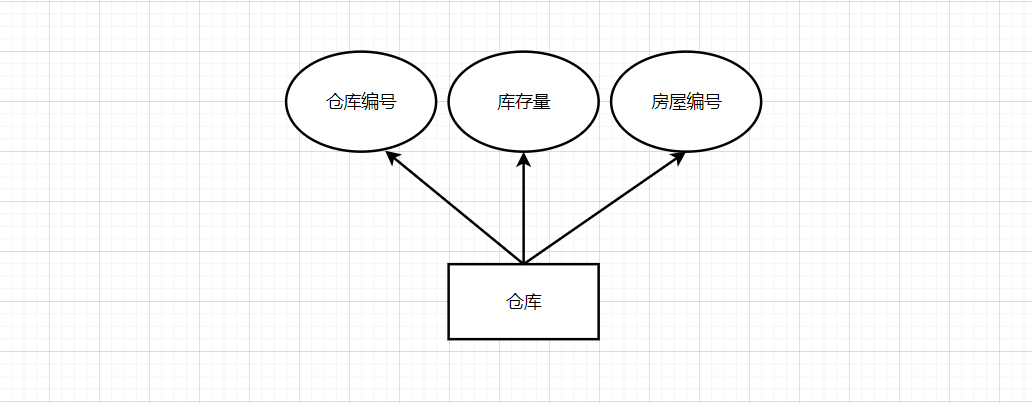
****

**分析：**

**房屋实体有以下五个主要属性列：房屋编号（主键，自增约束）、房屋名字、房屋地址、房主信息、房屋所需要的租金**

* + - 1. **房屋仓库管理实体（repo表）**

**ER图：**

****

* 1. ****属性分析****
     + 1. ****租客：****

**编号：租客的编号，自增约束，主键，用于唯一确定一个租客**

**姓名：姓名是租客的一个重要信息，租房成功后用于返回给房主**

**可用资金：租客能用于租房的资金**

* + - 1. ****房主：****

**编号：房主的编号，用来确认唯一的房主**

**姓名：房主的姓名**

**房屋编号：该房主所发布的房屋的编号**

**房主电话信息：用于联系房主的电话信息**

* + - 1. ****房屋：****

**房屋编号：房屋的编号，用来确定唯一的房屋**

**房名：房子的具体名称**

**房屋地址：房子所在的具体地址**

**房主信息：房屋的房主的信息**

**需要的租金：该房屋所需要的租赁价格**

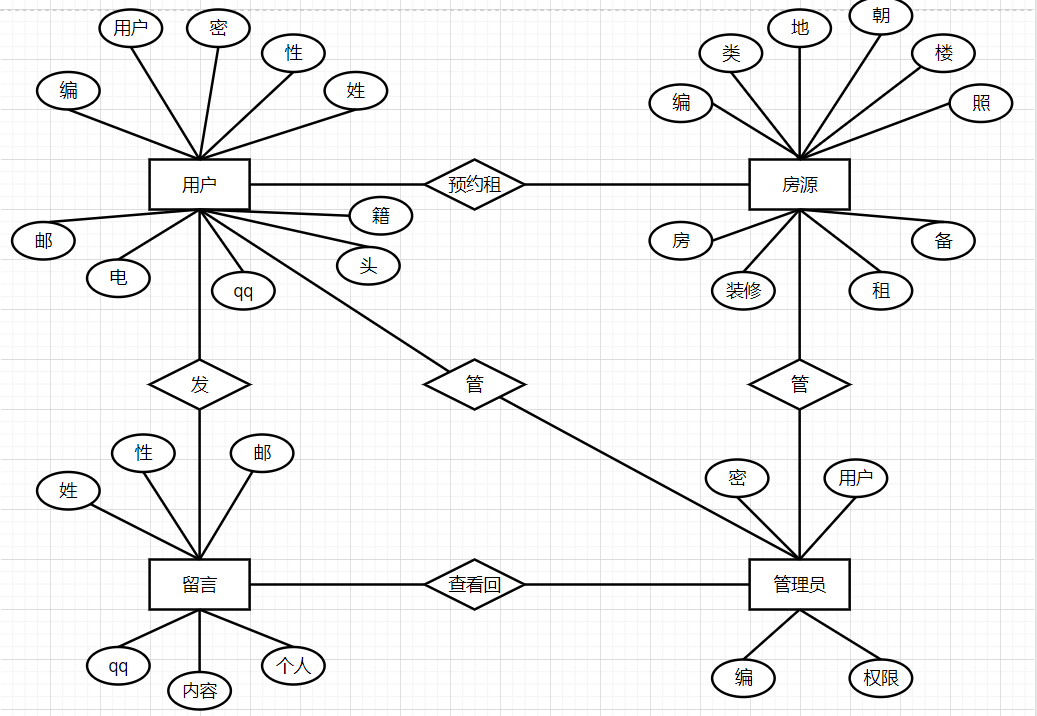
* + - 1. ****房屋仓库：****

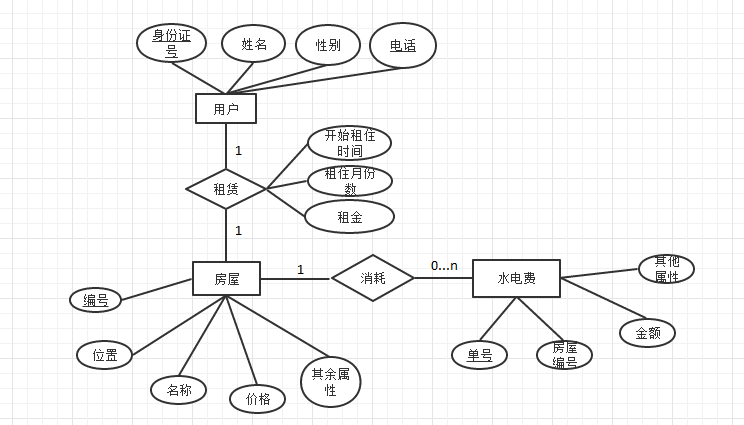
**仓库编号：仓库的编号，用来确认仓库**

**库存量：该仓库里剩下的房源**

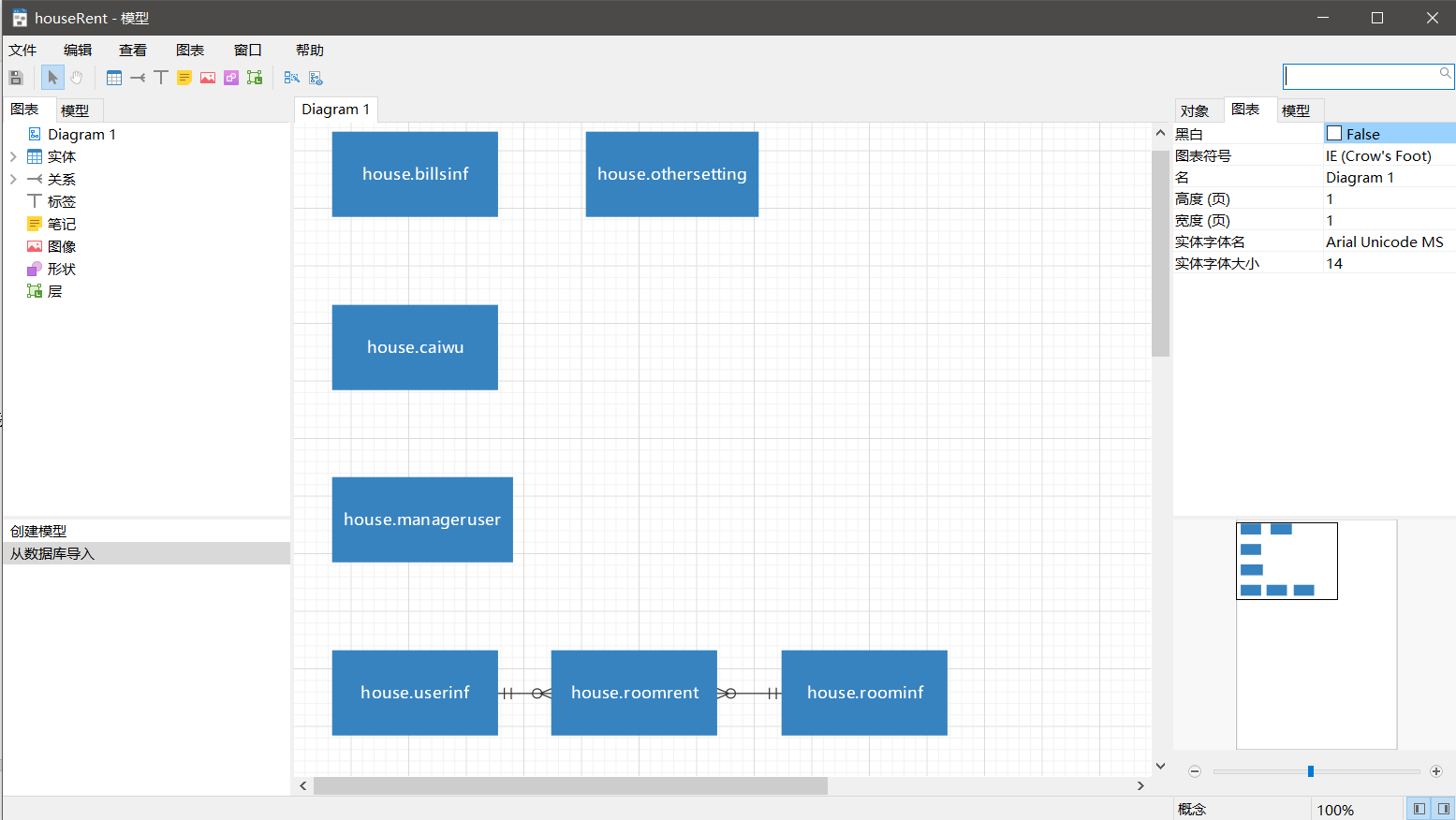
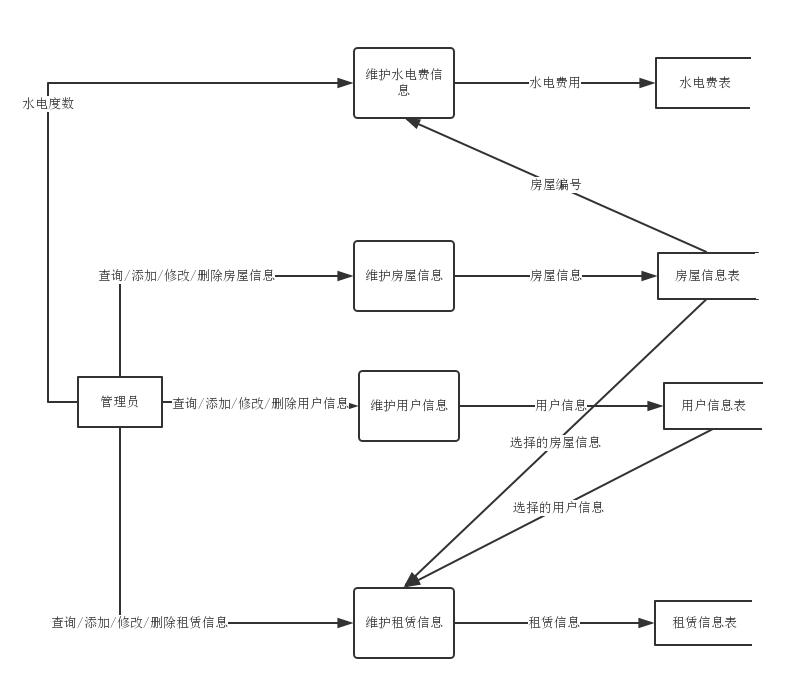
**房屋编号：仓库里房源的房子编号**

* 1. ****联系分析****

********



* 1. ****概念模型分析（.cdm图）****

********

# 数据库逻辑结构设计

关系模式（有下划线的是主键，所有表的数据设为varchar类型）：

管理员（用户名、密码）

房屋信息（房屋编号、所属区域、房屋名称、房屋类型、房屋面积、装修情况、房屋设施、房屋用途、月租价格、房间状态、备注说明）

用户信息（身份证号、用户姓名、性别、电话）

租赁信息（房屋编号、租户身份证号、租赁开始时间、租赁月数 、月租金）

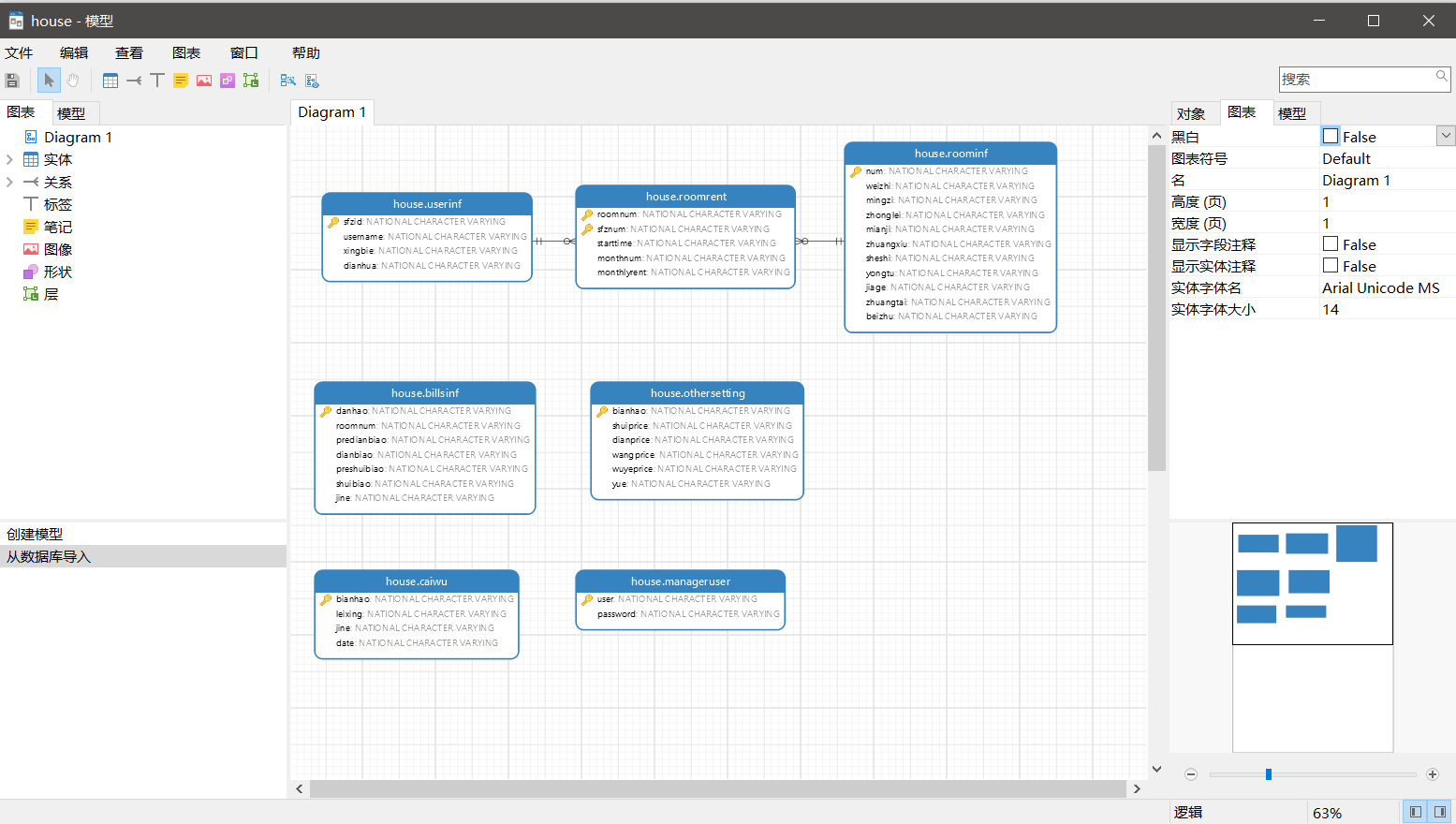
水电费信息（受理单号、房屋编号、上次电表数、本次电表数、上次水表数、本次水表数、收费金额）

财务收支报（编号、类型、金额、日期）

杂项设置（编号、水费单价、电费单价、宽带费单价、物业费单价、财务余额）

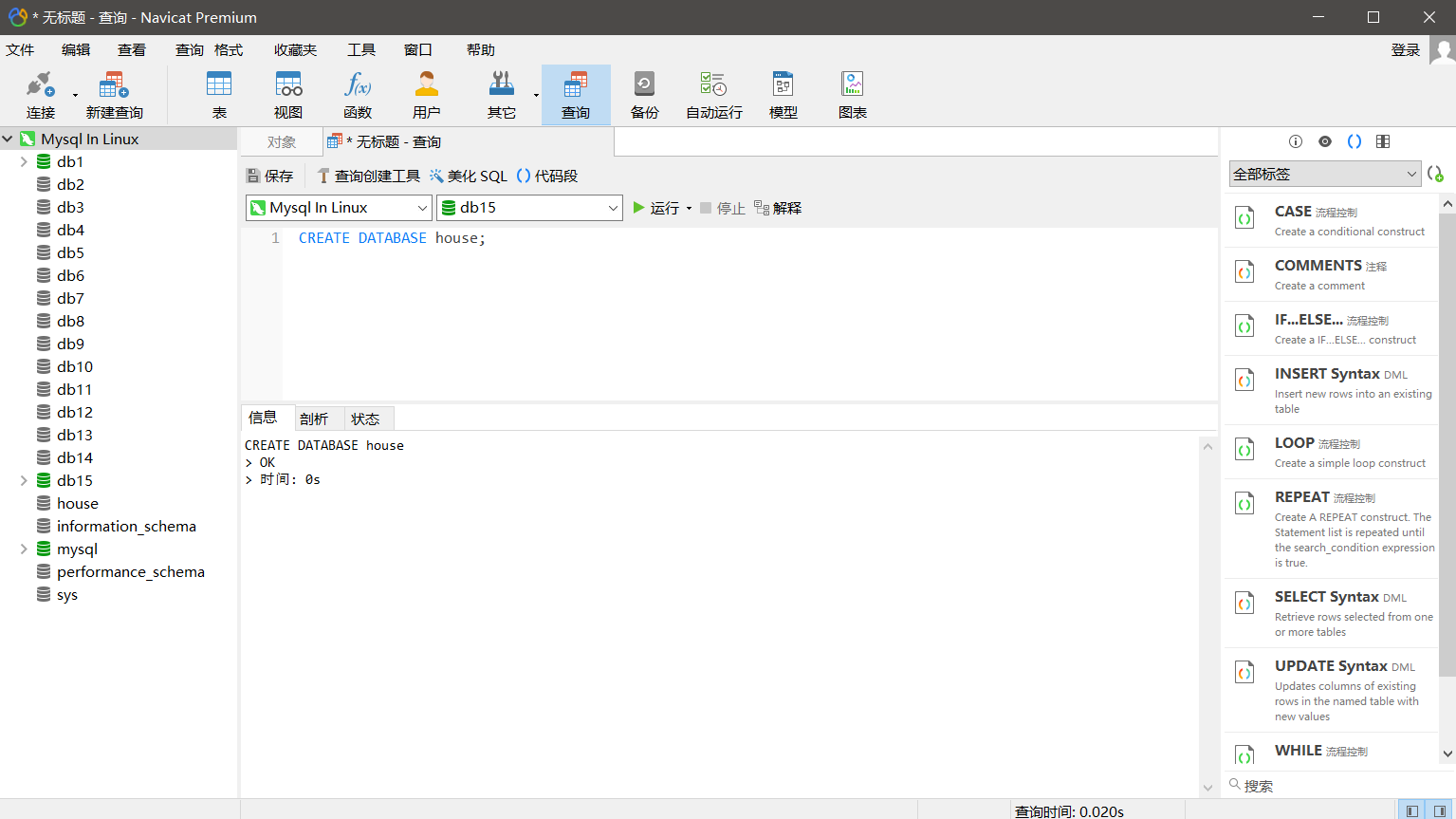
注：编号位数字等限制由程序中的函数完成，故所有数据类型均为varchar

**数据库逻辑模型图（.LDM）**

****

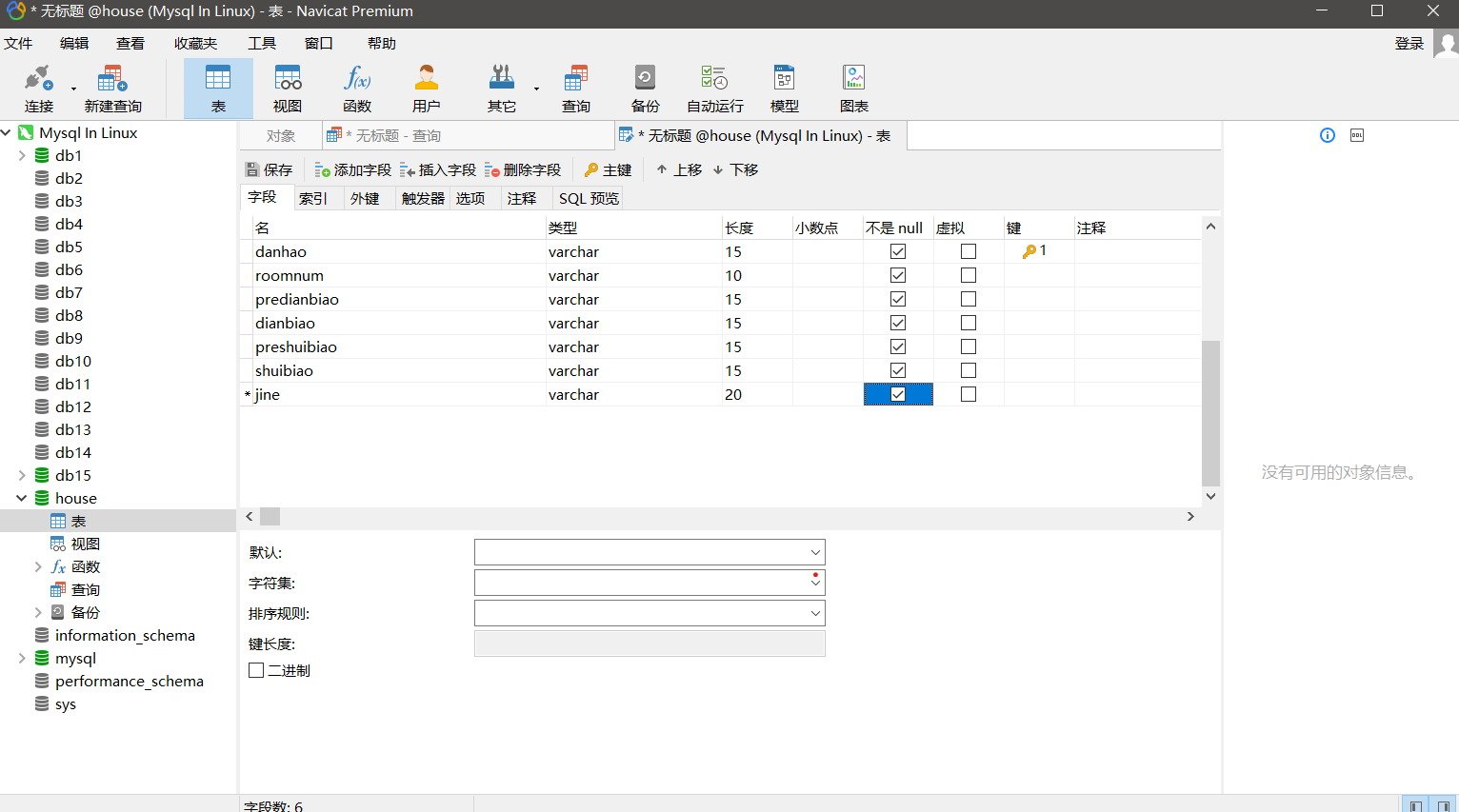
# 数据库物理实现

1. **数据库的创立：**

****

1. **数据表的物理实现：**

**1.水电费表：以下为Navicat上的建表过程**



**导出的SQL语句：**

**CREATE TABLE IF NOT EXISTS `billsinf` (**

**`danhao` varchar(15) COLLATE utf8\_unicode\_ci NOT NULL,**

**`roomnum` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,**

**`predianbiao` varchar(15) COLLATE utf8\_unicode\_ci NOT NULL,**

**`dianbiao` varchar(15) COLLATE utf8\_unicode\_ci NOT NULL,**

**`preshuibiao` varchar(15) COLLATE utf8\_unicode\_ci NOT NULL,**

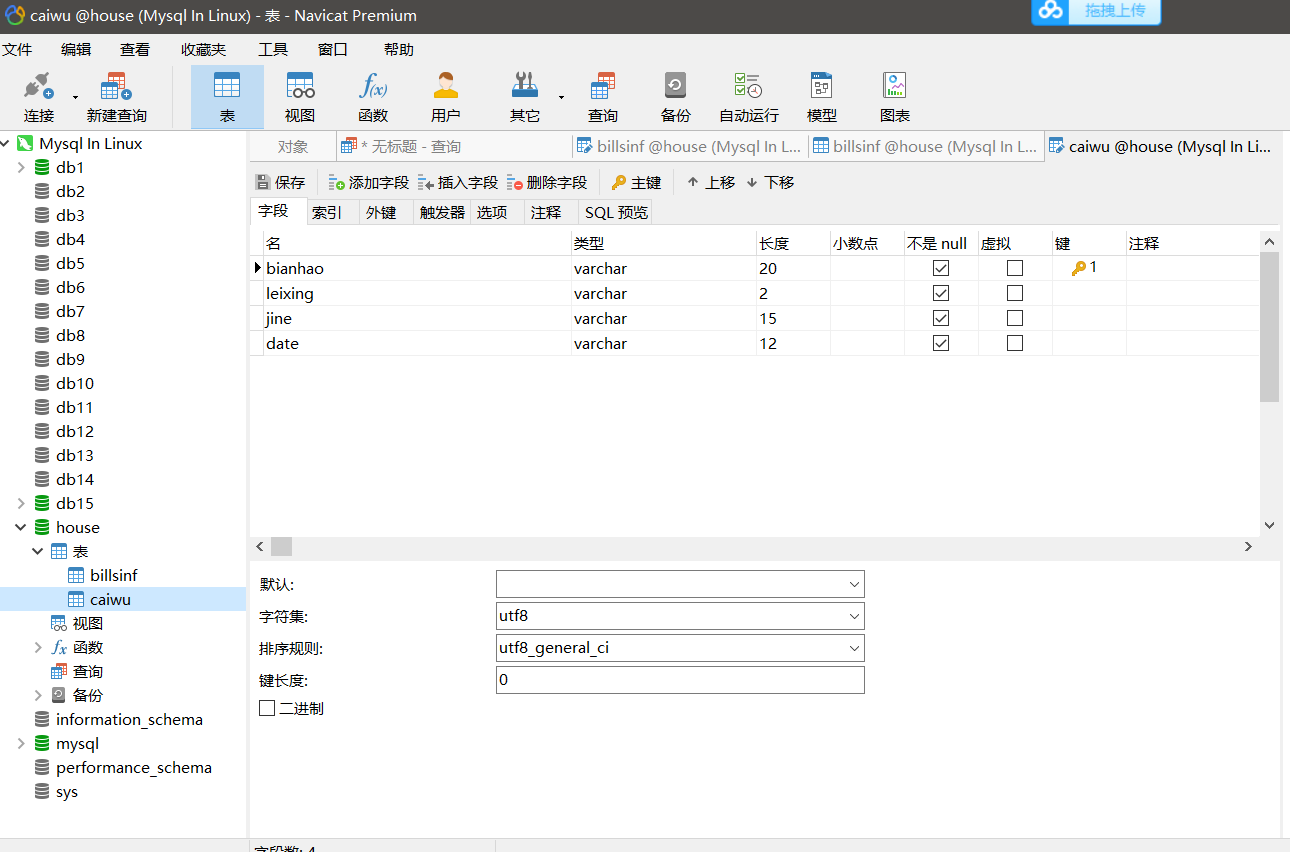
**`shuibiao` varchar(15) COLLATE utf8\_unicode\_ci NOT NULL,**

**`jine` varchar(20) COLLATE utf8\_unicode\_ci NOT NULL,**

**PRIMARY KEY (`danhao`)**

**) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;**

**2.财务表：**

****

**SQL语句：**

CREATE TABLE IF NOT EXISTS `caiwu` (

`bianhao` varchar(20) COLLATE utf8\_unicode\_ci NOT NULL,

`leixing` varchar(2) COLLATE utf8\_unicode\_ci NOT NULL,

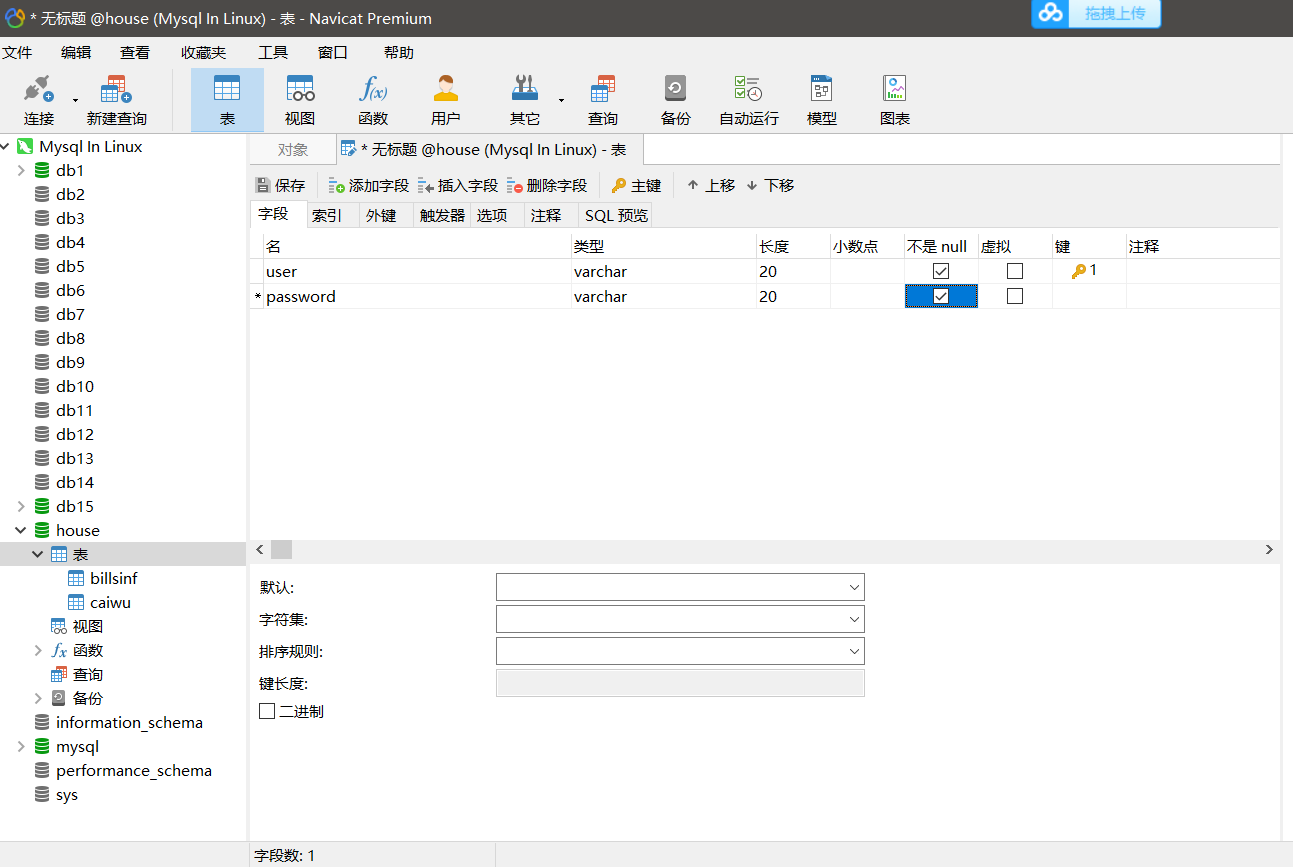
`jine` varchar(15) COLLATE utf8\_unicode\_ci NOT NULL,

`date` varchar(12) COLLATE utf8\_unicode\_ci NOT NULL,

PRIMARY KEY (`bianhao`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

**3.管理员表：**

****

**SQL语句：**

CREATE TABLE IF NOT EXISTS `manageruser` (

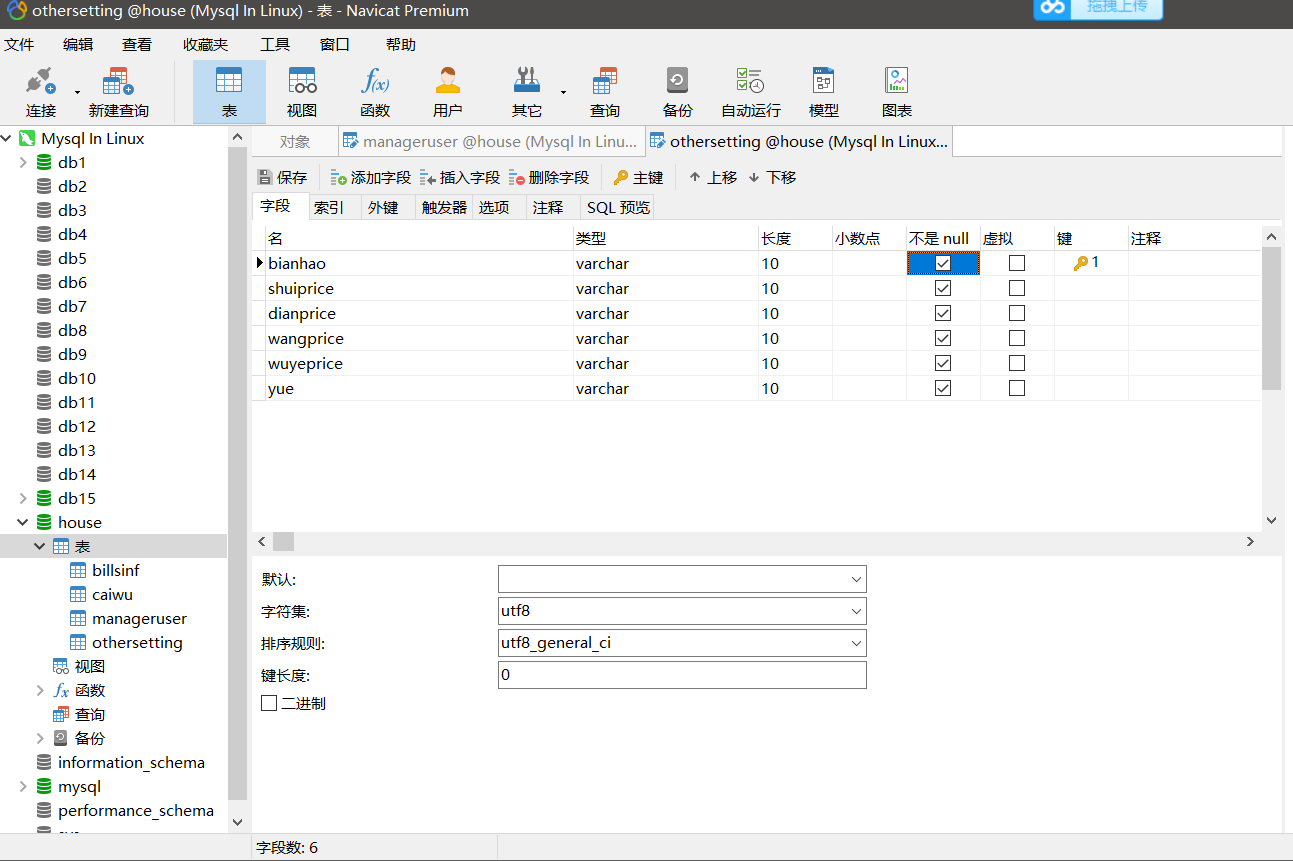
`user` varchar(20) COLLATE utf8\_unicode\_ci NOT NULL,

`password` varchar(20) COLLATE utf8\_unicode\_ci NOT NULL,

PRIMARY KEY (`user`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

**4.杂项表：**

****

**SQL语句：**

CREATE TABLE IF NOT EXISTS `othersetting` (

`bianhao` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,

`shuiprice` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,

`dianprice` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,

`wangprice` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,

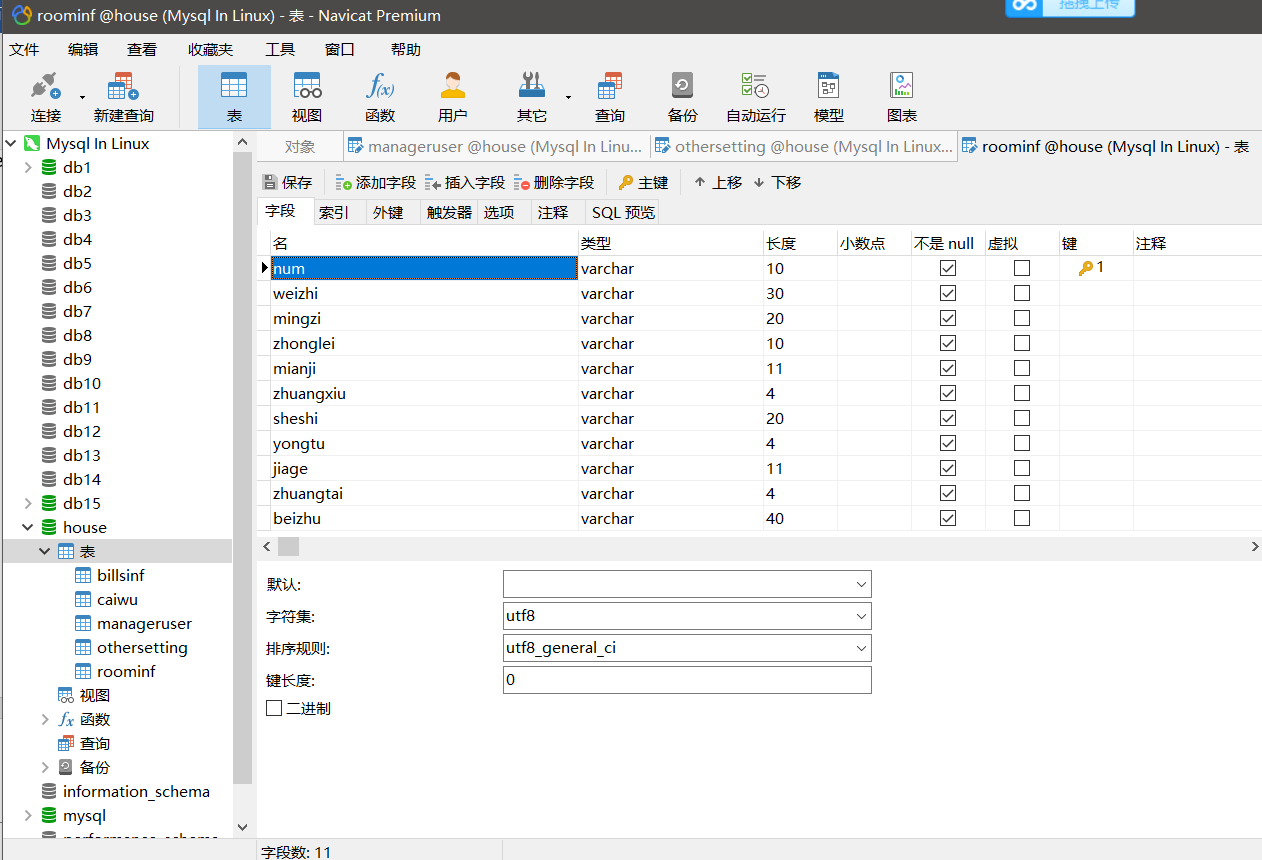
`wuyeprice` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,

`yue` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,

PRIMARY KEY (`bianhao`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

**5.房屋信息表：**

****

**SQL语句：**

CREATE TABLE IF NOT EXISTS `roominf` (

`num` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,

`weizhi` varchar(30) COLLATE utf8\_unicode\_ci NOT NULL,

`mingzi` varchar(20) COLLATE utf8\_unicode\_ci NOT NULL,

`zhonglei` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,

`mianji` varchar(11) COLLATE utf8\_unicode\_ci NOT NULL,

`zhuangxiu` varchar(4) COLLATE utf8\_unicode\_ci NOT NULL,

`sheshi` varchar(20) COLLATE utf8\_unicode\_ci NOT NULL,

`yongtu` varchar(4) COLLATE utf8\_unicode\_ci NOT NULL,

`jiage` varchar(11) COLLATE utf8\_unicode\_ci NOT NULL,

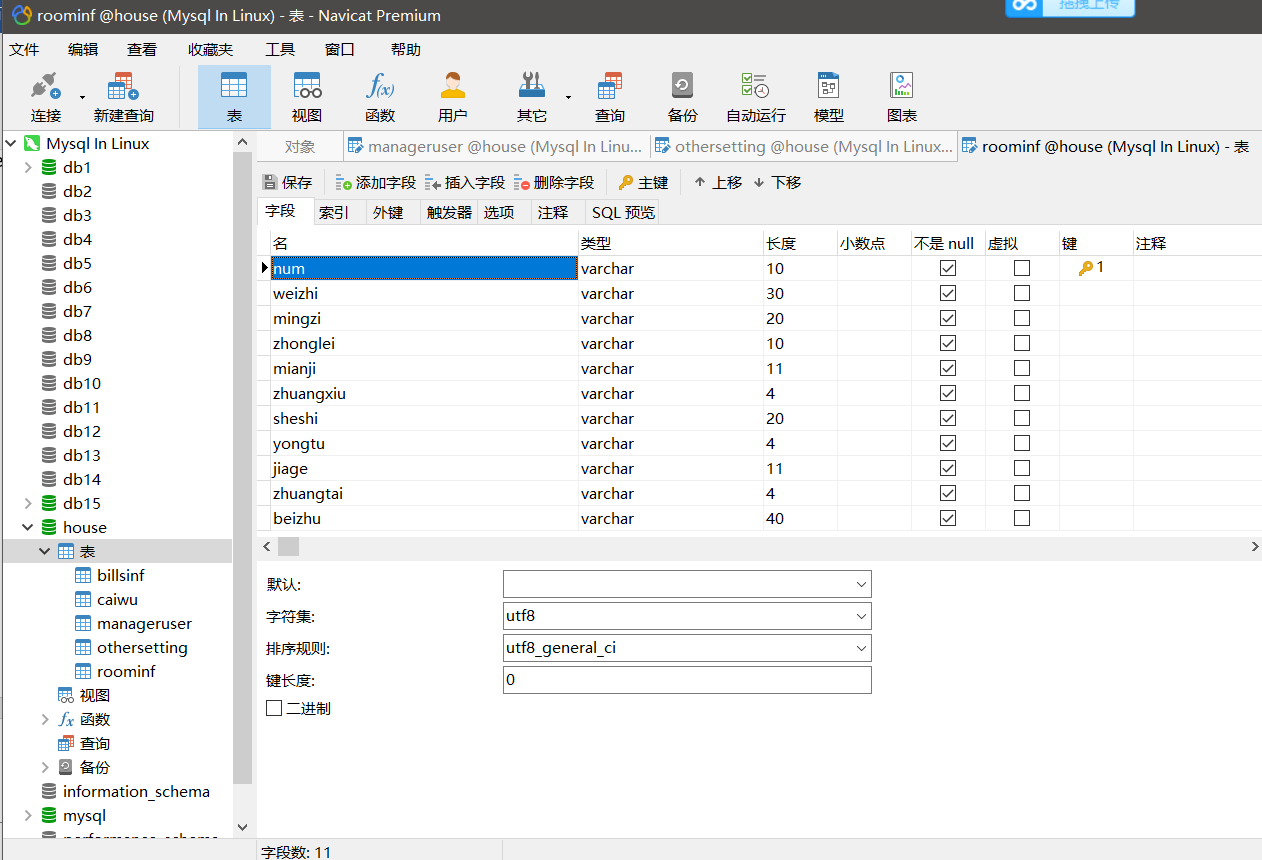
`zhuangtai` varchar(4) COLLATE utf8\_unicode\_ci NOT NULL,

`beizhu` varchar(40) COLLATE utf8\_unicode\_ci NOT NULL,

PRIMARY KEY (`num`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

**6.房屋租赁表：**

****

**SQL语句：**

CREATE TABLE IF NOT EXISTS `roomrent` (

`roomnum` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,

`sfznum` varchar(18) COLLATE utf8\_unicode\_ci NOT NULL,

`starttime` varchar(12) COLLATE utf8\_unicode\_ci NOT NULL,

`monthnum` varchar(4) COLLATE utf8\_unicode\_ci NOT NULL,

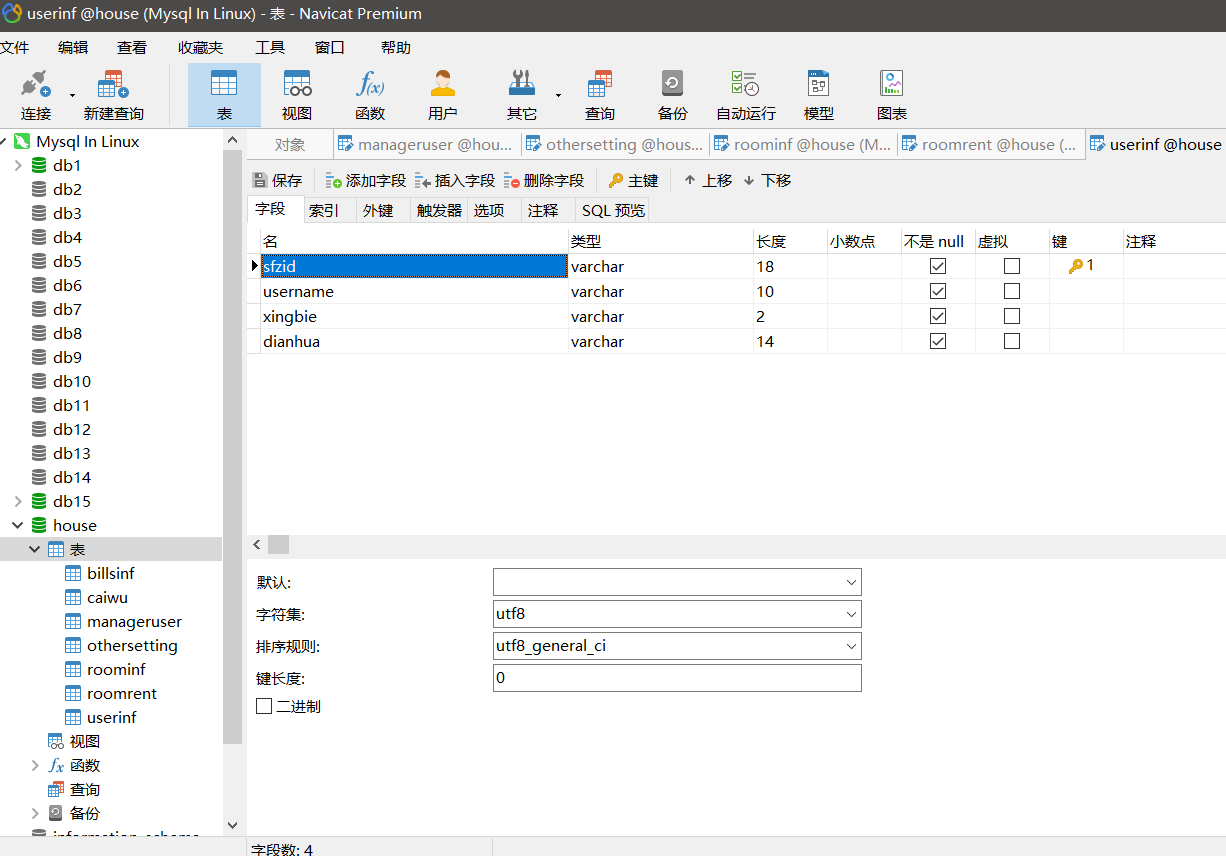
`monthlyrent` varchar(20) COLLATE utf8\_unicode\_ci NOT NULL,

PRIMARY KEY (`roomnum`,`sfznum`),

KEY `userinfyueshu` (`sfznum`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

**7.用户信息表：**

****

**SQL语句：**

CREATE TABLE IF NOT EXISTS `userinf` (

`sfzid` varchar(18) COLLATE utf8\_unicode\_ci NOT NULL,

`username` varchar(10) COLLATE utf8\_unicode\_ci NOT NULL,

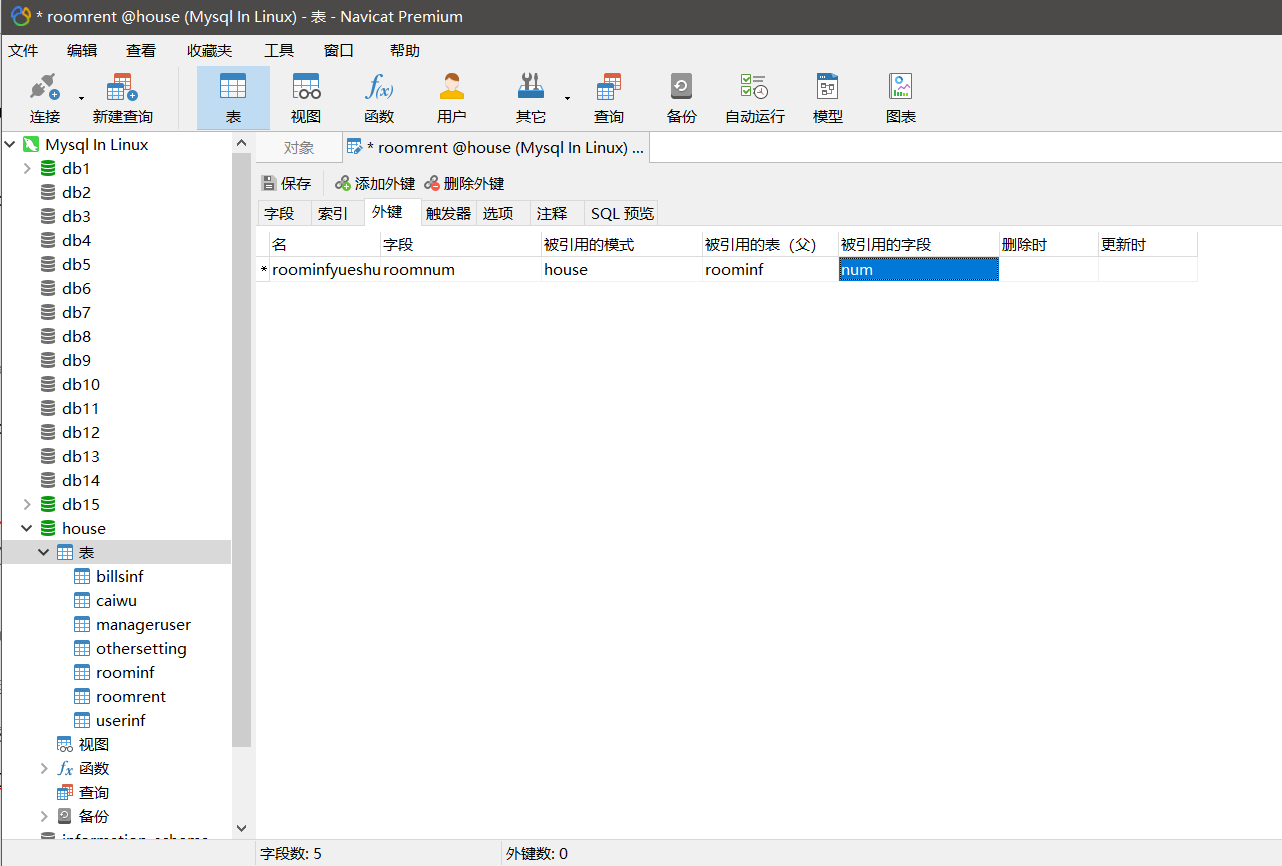
`xingbie` varchar(2) COLLATE utf8\_unicode\_ci NOT NULL,

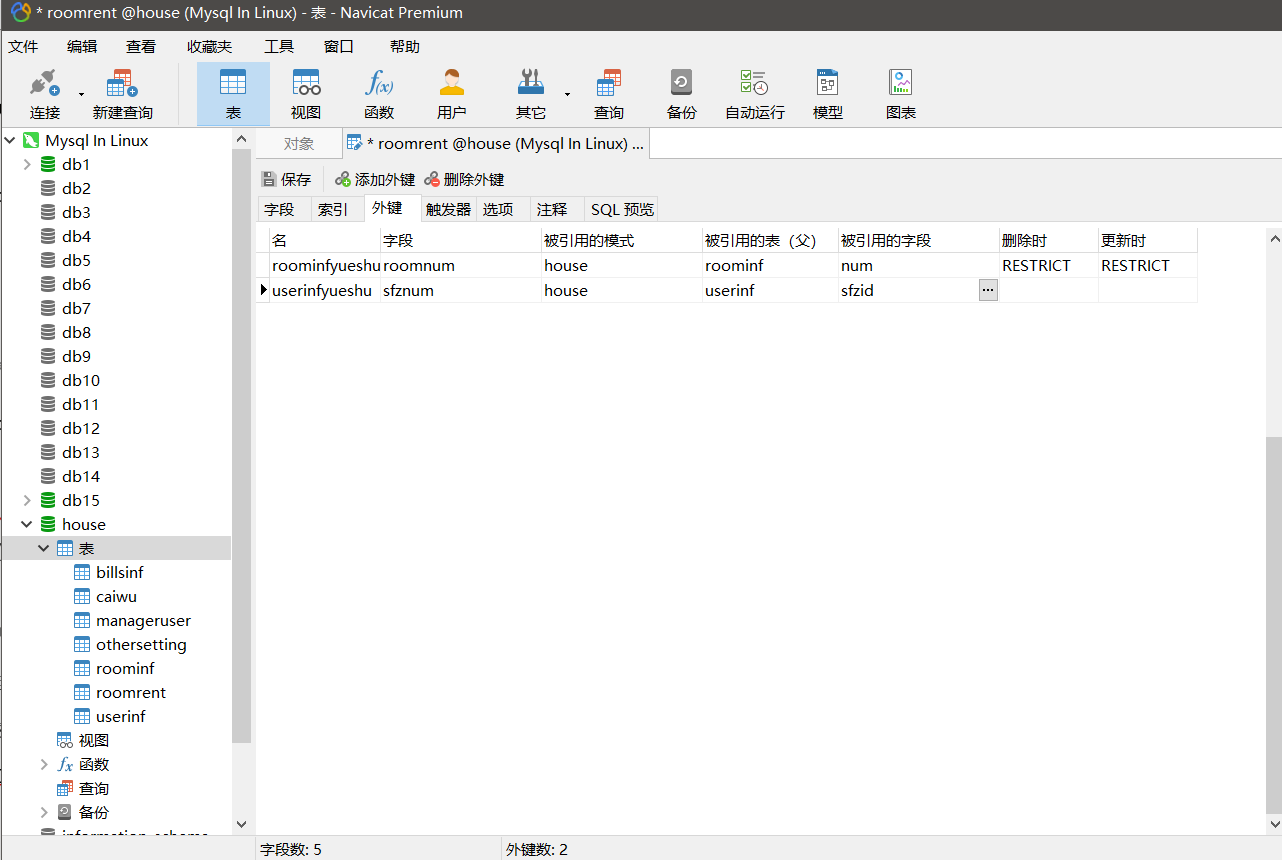
`dianhua` varchar(14) COLLATE utf8\_unicode\_ci NOT NULL,

PRIMARY KEY (`sfzid`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

1. **数据库约束关系实现：**

****

****

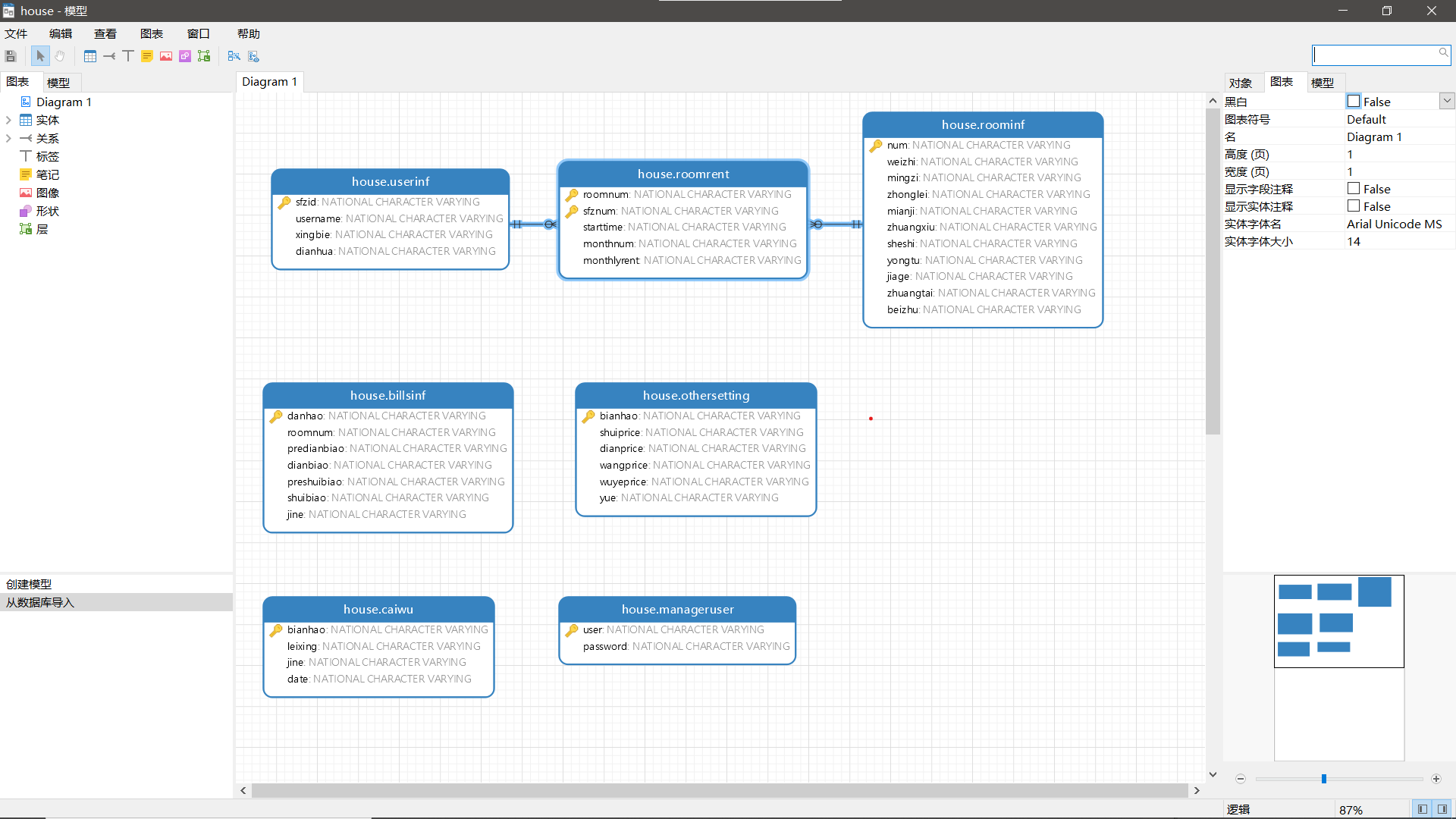
**SQL语句：**

ALTER TABLE `roomrent`

ADD CONSTRAINT `roominfyueshu` FOREIGN KEY (`roomnum`) REFERENCES `roominf` (`num`) ON DELETE CASCADE ON UPDATE CASCADE,

ADD CONSTRAINT `userinfyueshu` FOREIGN KEY (`sfznum`) REFERENCES `userinf` (`sfzid`) ON DELETE CASCADE ON UPDATE CASCADE;

**（5）数据库物理模型图：**

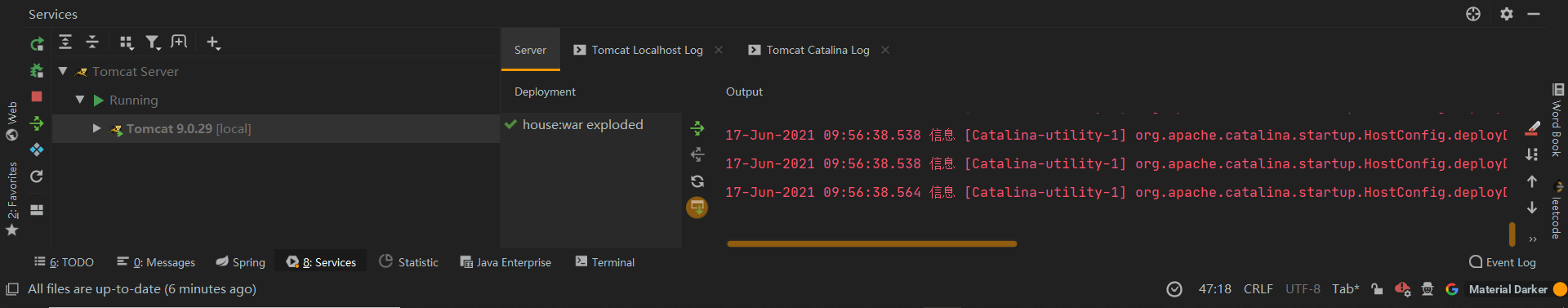
****

# 数据库功能调试

1. **JavaWeb开发：**

为了使用这次开发的数据库，需要编写一个建议的javaweb系统对数据库进行调用，这次我编写的系统是基于SSM框架技术编写的系统软件。SSM：S（spring），S（springcloud），M（mybatis）。在这次大作业中，主要偏向于数据库的连接调用，所以我主要集中讲解mybatis的实现和项目的部署。

1. **IDEA集成TomCat部署项目：**

****

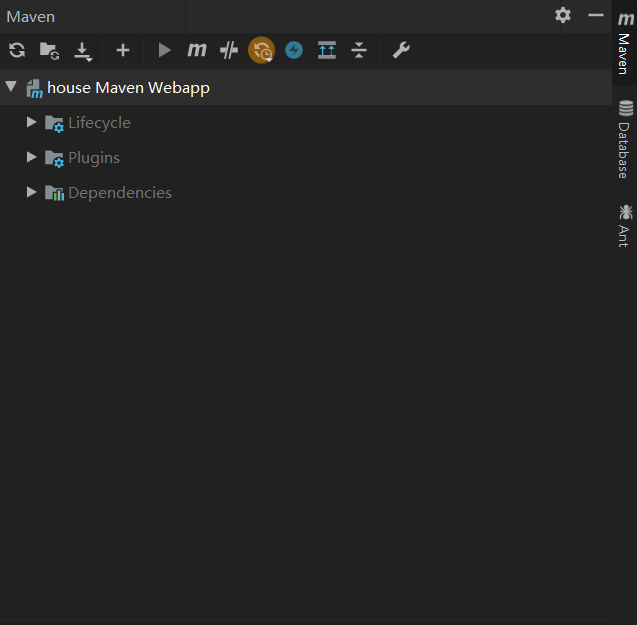
**在windows下配置好tomcat后集成在idea用于部署发布应用，点击运行，部署成功后将会自动打开index.jsp文件路径在默认得浏览器（默认端口号为8080）**

1. **项目的创建：**

创建maven项目，这里简单介绍一下maven项目。

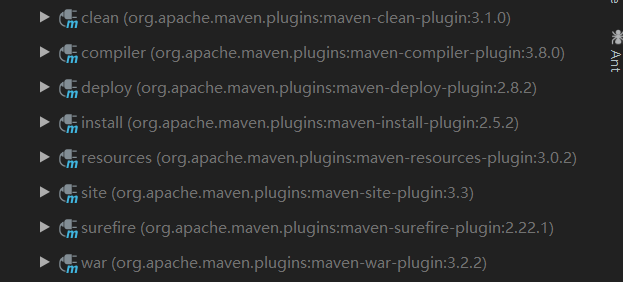
Maven项目本质意义就像是一个仓库，它帮我们保管了本次项目所用的jar包、依赖和一些插件（在pom.xml下）

Maven项目的管理页面：

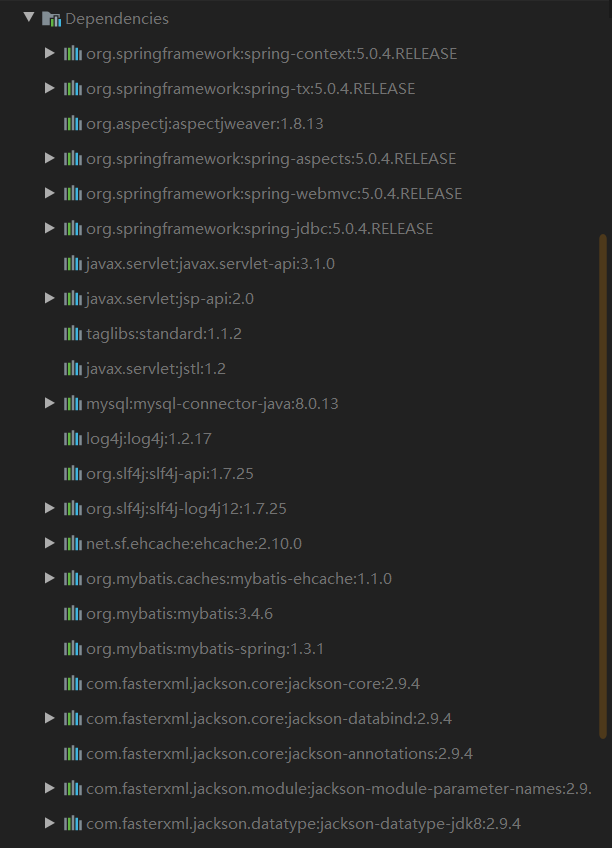
****

Lifecyle：项目的生命周期

Plugins：项目中用到的一些插件如分页插件，tomcat插件等

****

Dependencies：项目中用到的依赖关系

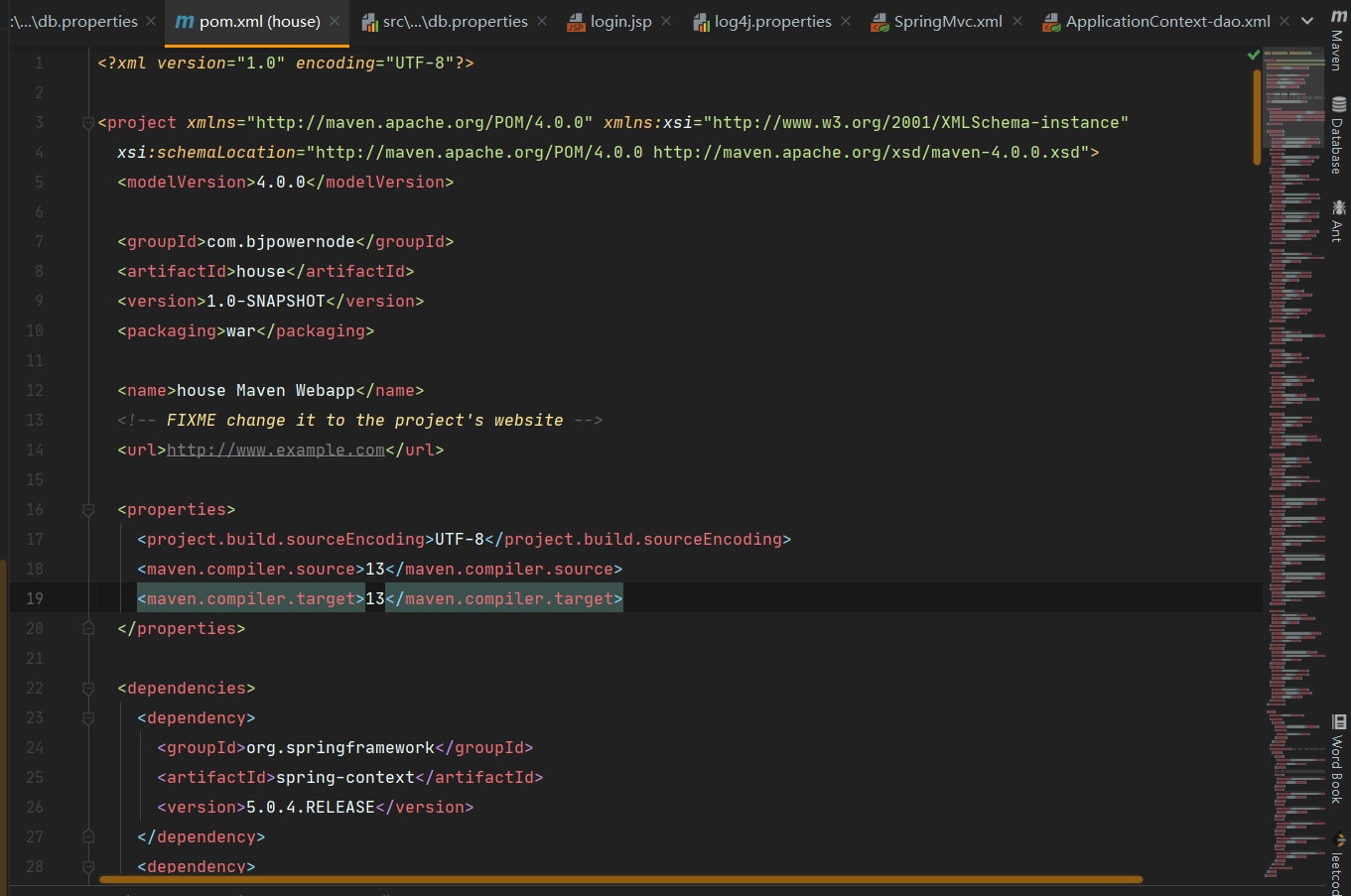


包含了项目中我所用的jar包

重要的有：mybatis包（内涵JDBC框架）用于连接数据库，以及javax内含javeEE下的httpservlet用于网页段的和后端的连接开发。

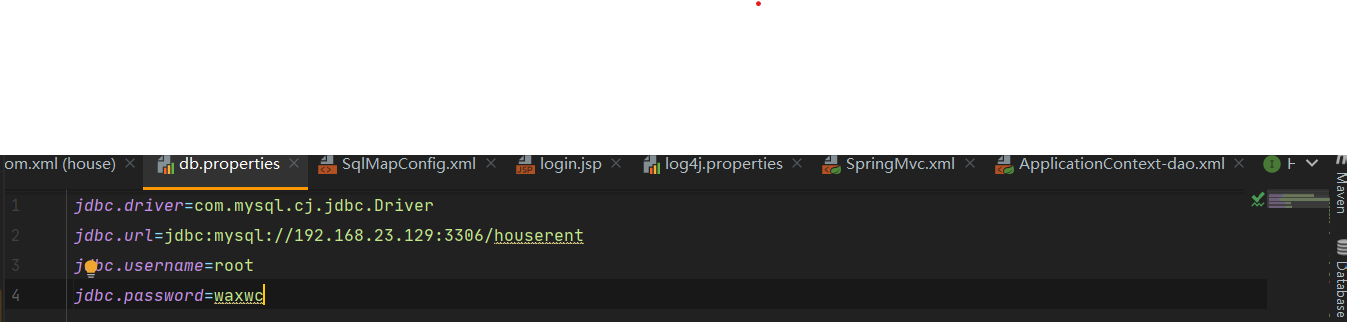
**Pom.xml文件：**

这是maven工程的核心配置文件，包含了你所要使用的所有的依赖编写。



1. **用于连接数据库的mybatis编写**

Mybatis是一个基于javaweb开发持久层的框架（软件半成品），里面包含了用于连接数据库的JDBC功能。所以这是我用来调用mysql数据库的主要配置文件和功能。



Mybatis连接数据库的主要配置文件db.properies

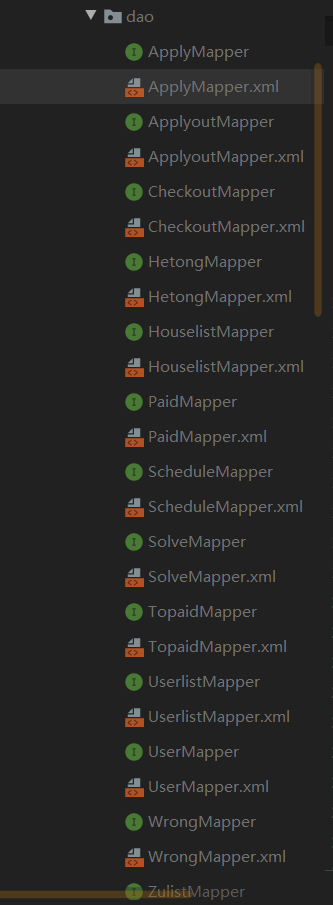
里面主要调用类jdbc.driver(用来开启driver类实现连接池连接)

Jdbc.url:这里面写的是我要链接的数据库是mysql数据库，以及主机的名字（ip号）以及端口号（默认3306）后面就是我要链接的数据库houserent

Jdbc.username：我所用数据库管理系统所在的用户名（这里我用的是linux系统）所以在根目录（root）下。

Jdbc.password：我的数据库管理系统的密码（waxwc）。

1. **编写dao层的mapper文件，实现数据库的使用**

****

Dao层就是javaweb开发层次里的最底层，也叫持久层，它可以实现项目与数据库的直接交互。我将它理解为数据库系统里的数据库管理人员它可以直接对数据库里的数据进行交互和调用。

主要内容：

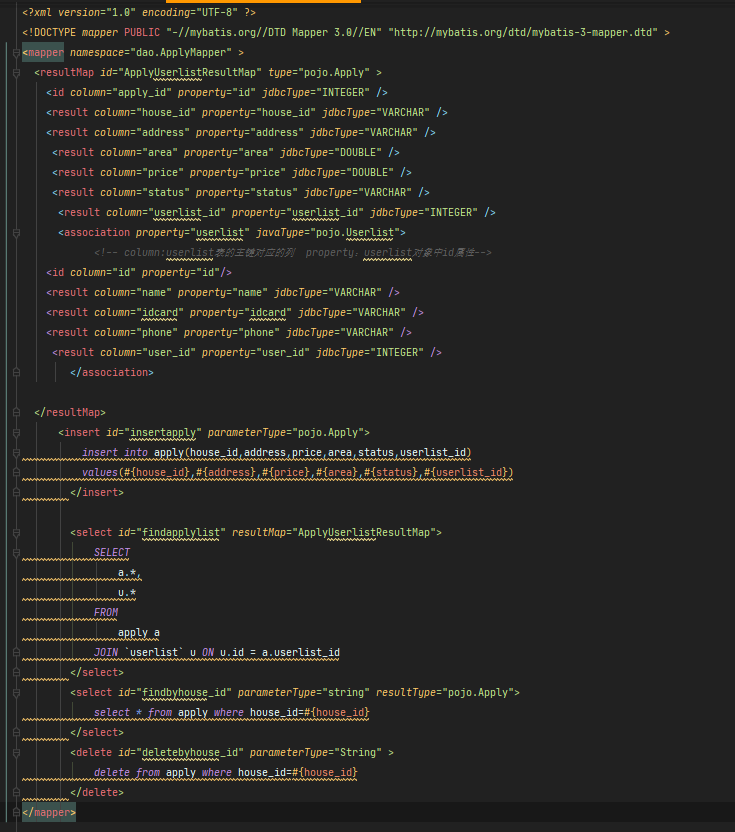
Mapper文件：这是对各功能部件和数据库调用实现功能的主要文件，用于写SQL语句实现功能。



各功能的接口文件

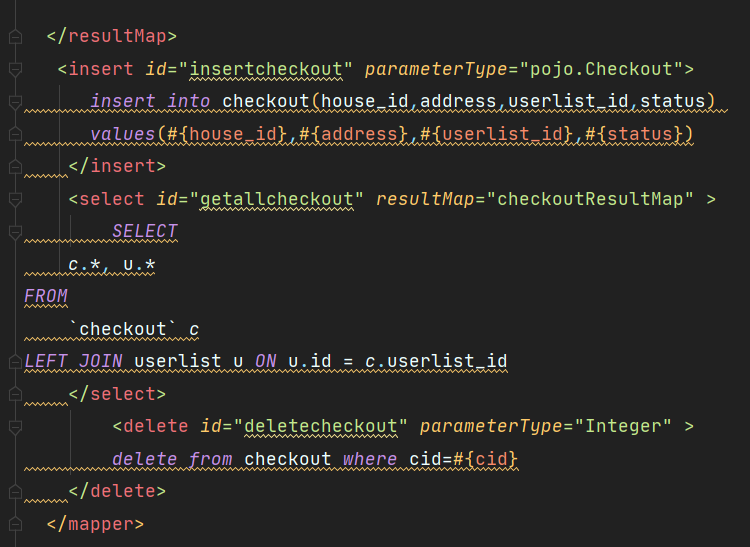


1. **登录功能的mapper实现：**

****

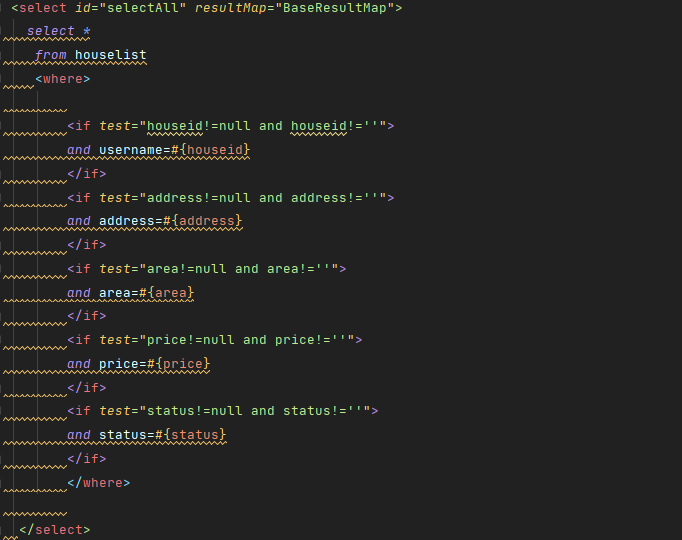
在数据库中查询存在的账号密码，符合的在前端进行页面的跳转。

1. **Checkout检测房源**

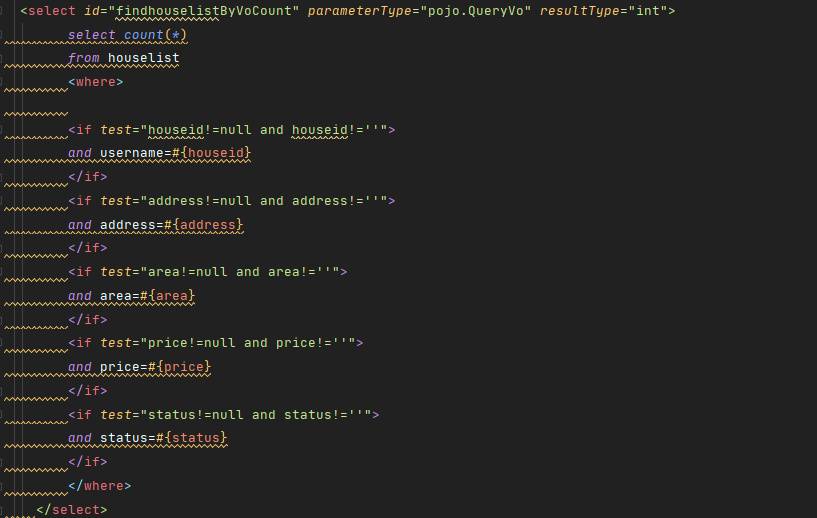


1. **房源展示mapper：**

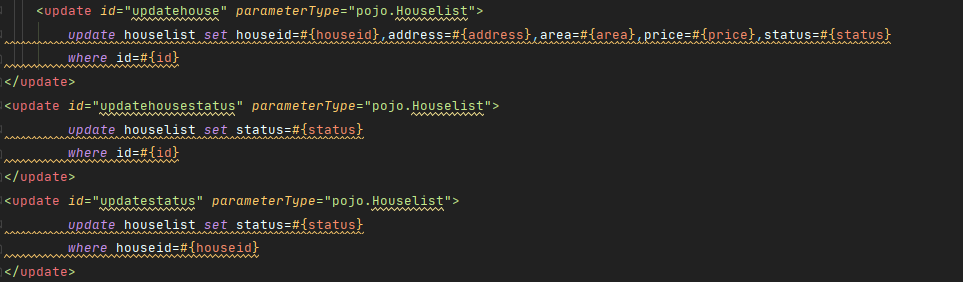
查询全部：

****

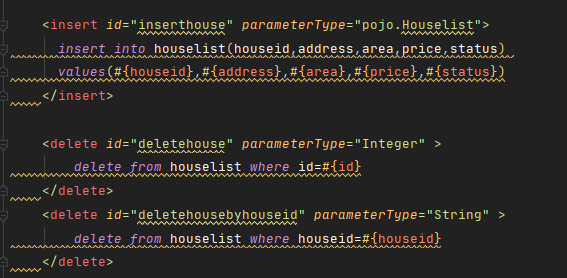
按id查询：



更新房屋数据：

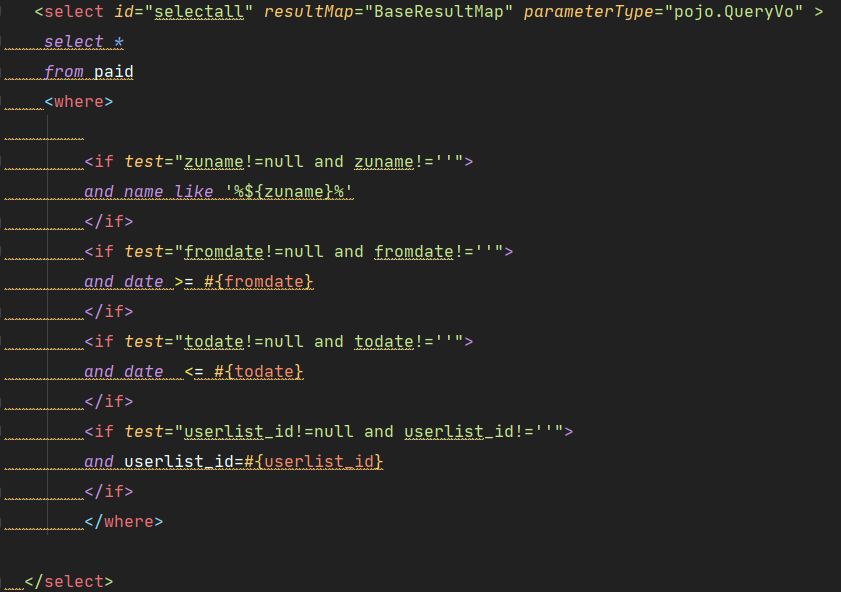


插入房屋数据：

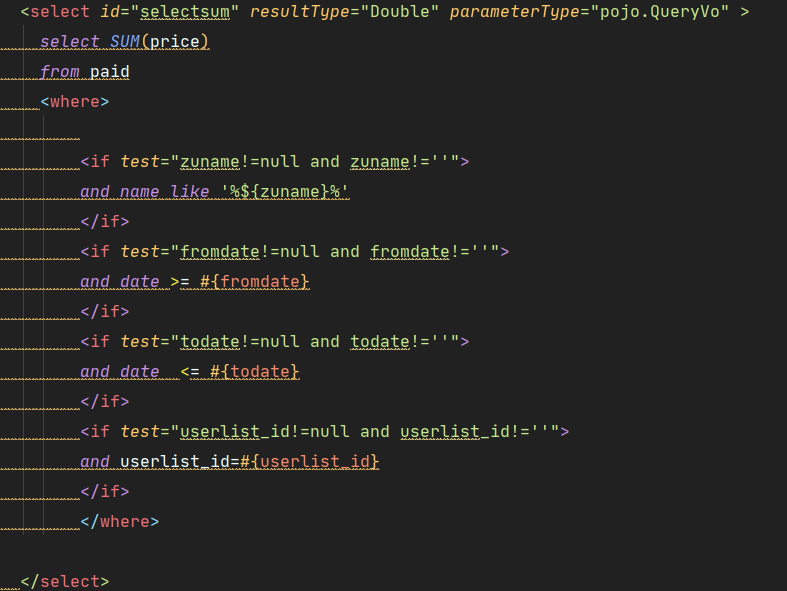


1. **开始租赁房屋mapper（租客租赁）**

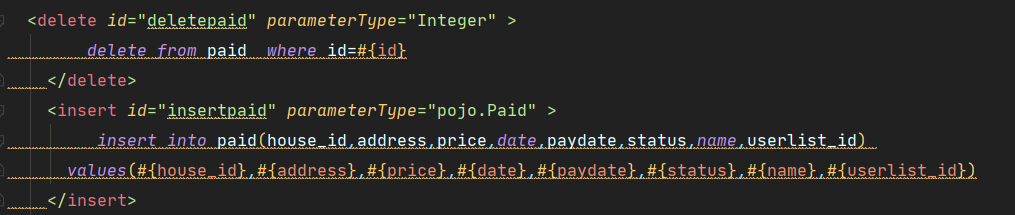
查询再租列表：



查询租客名字信息：



删除订单：

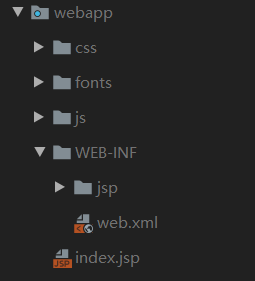


1. **用户设置：**

****

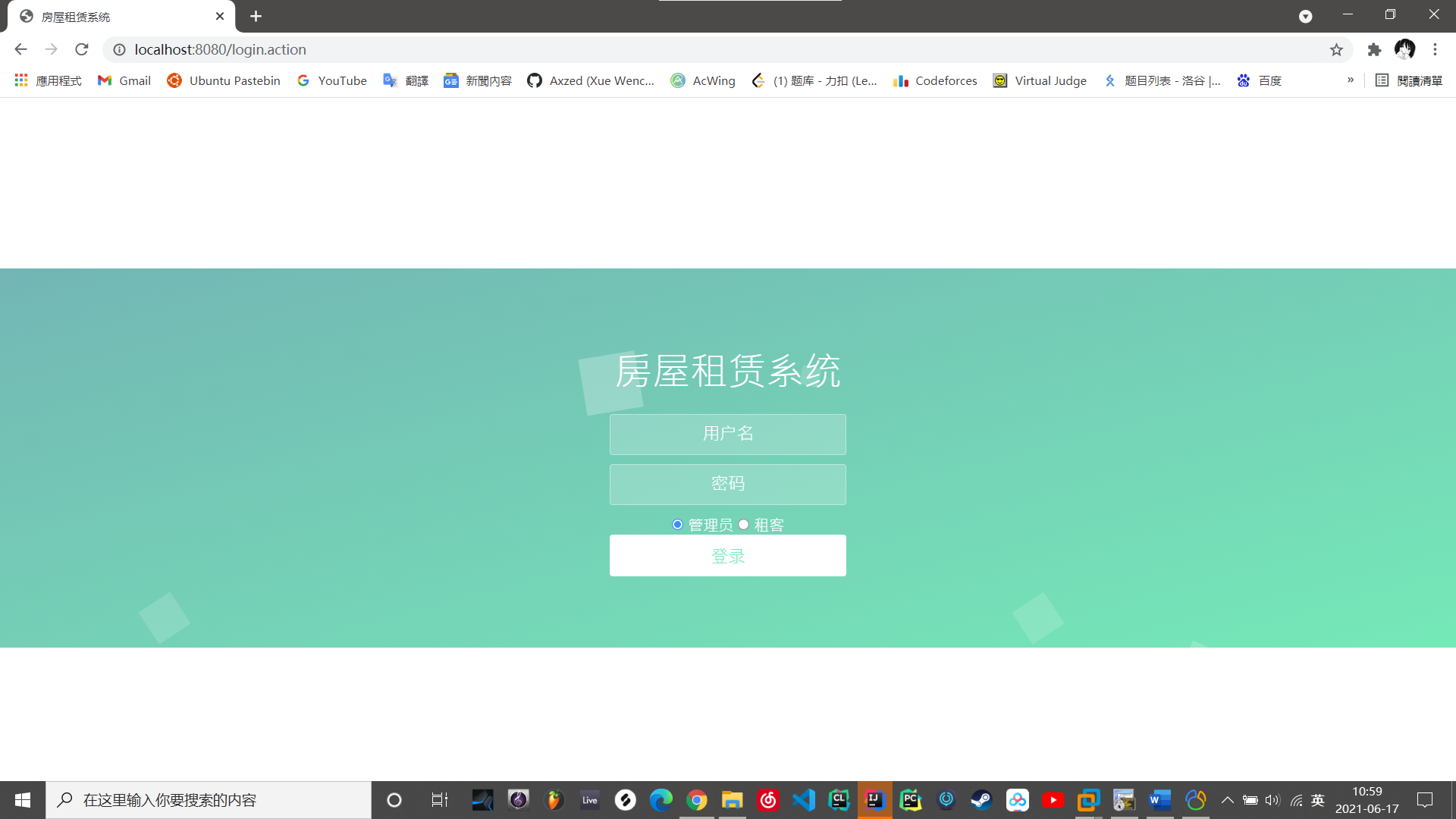
1. **前端页面基础搭建：**

基本框架：jsp，jquery，html，css，js

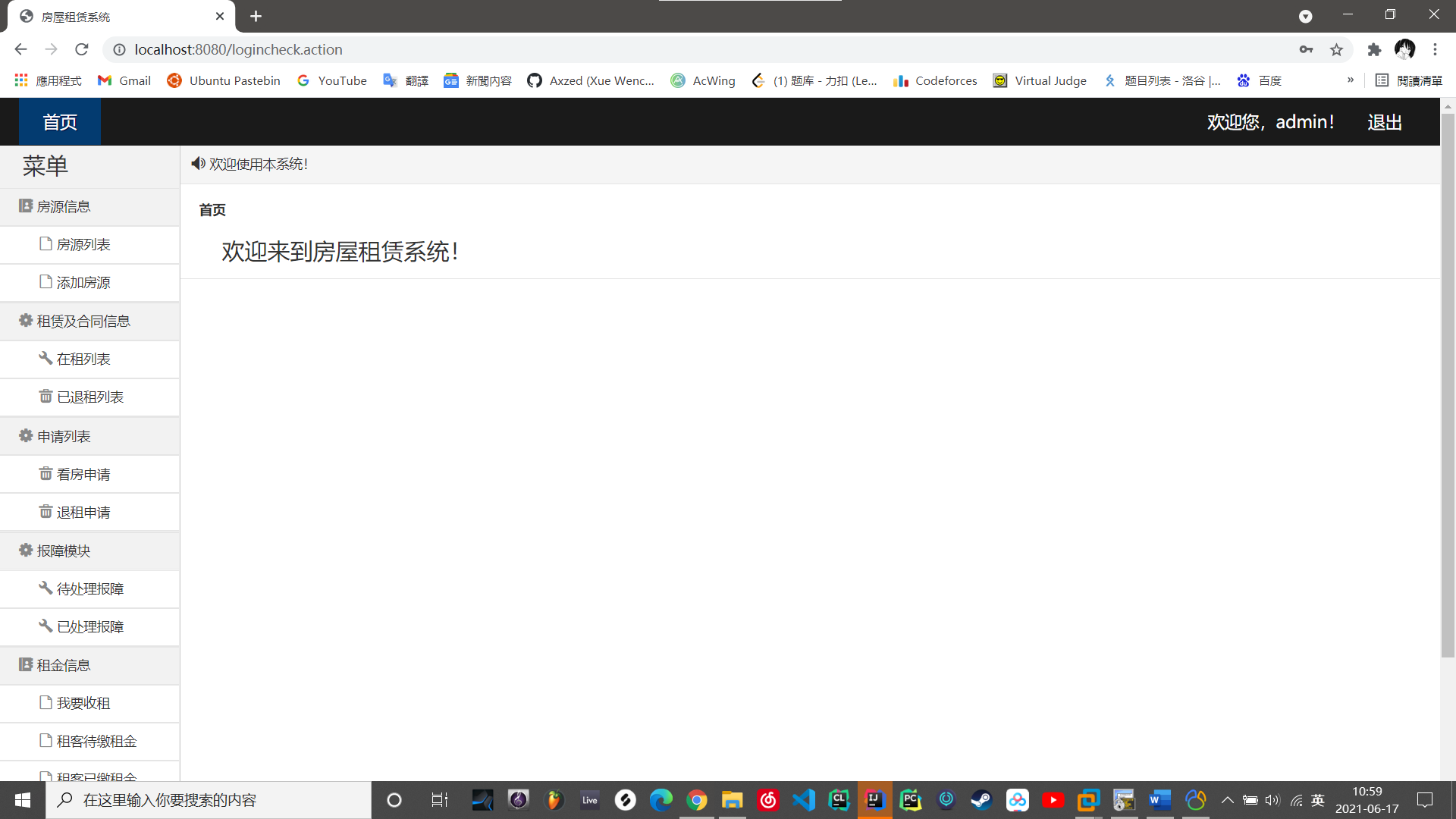


1. **项目展示：**

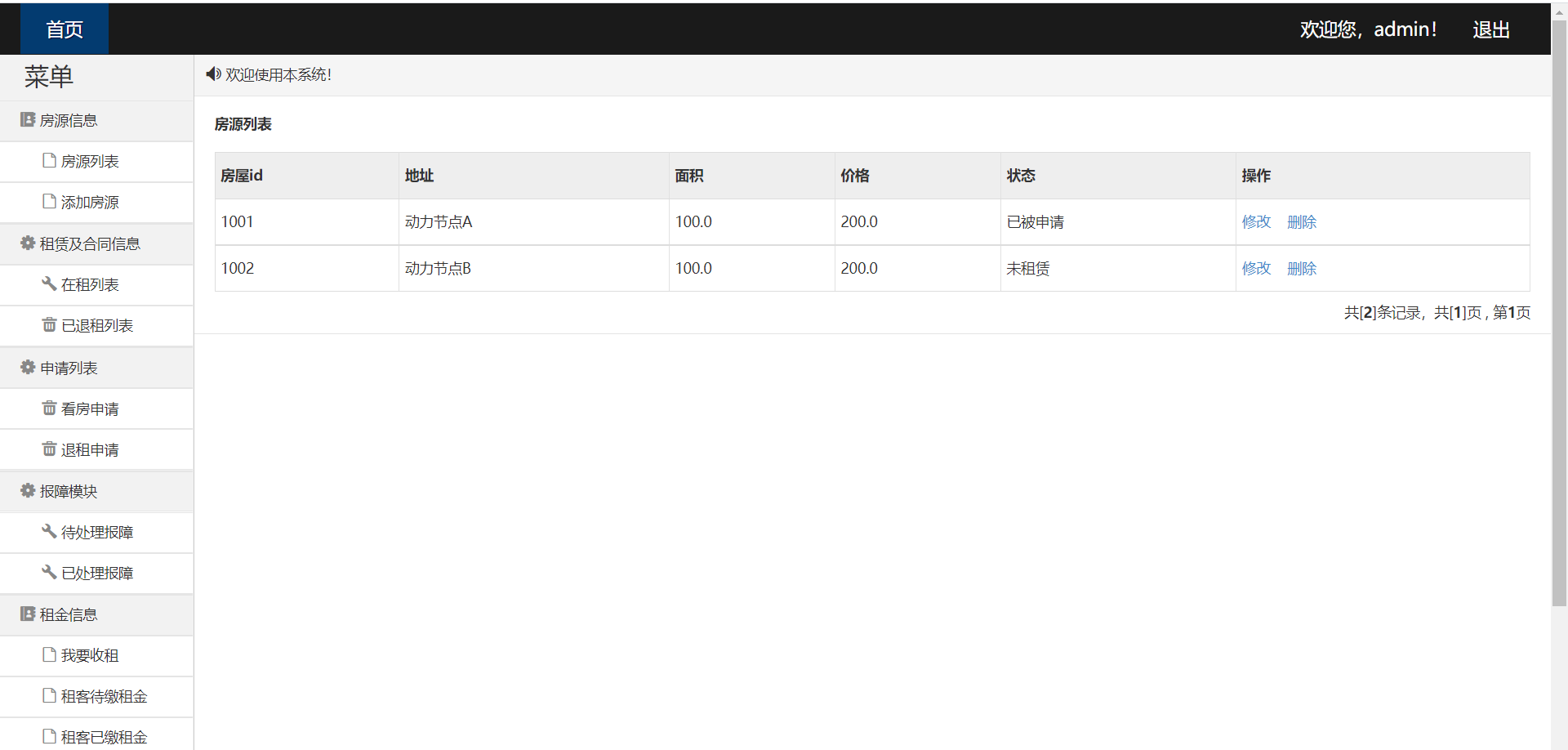
登录页面：



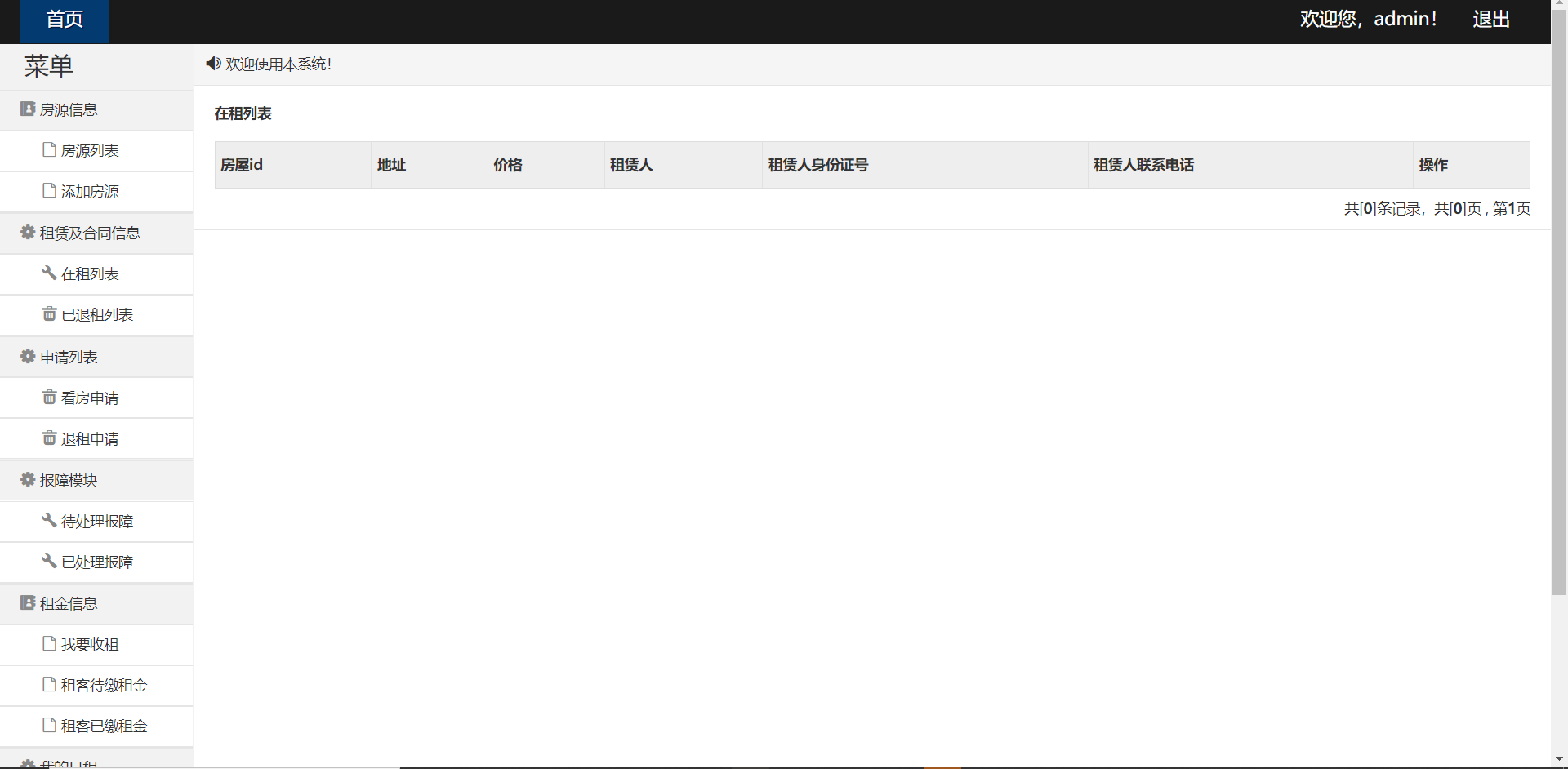
服务页面：



房屋列表：



在租列表：



# 设计总结

在这次实验中，有很多功能由于我的技术功能有限所以没有实现，在前期搭建数据库的时候我有很多的想法，但后期在项目设计的时候没能够用上，所以导致在表的设计上仍有很多的缺陷。

在功能模块的实现上，前端的框架是借用了github上的开源框架，内容靠自定义即可。

在连接了数据库之后，编写mapper和javaweb的三层框架是最为困难的，导致调试废案很多次，这也督促了我多去写一些项目。

综上所述，在历经一个多月的时间，从javaweb零基础到现在堪堪入门，这个项目终于是做好了。