# 目录

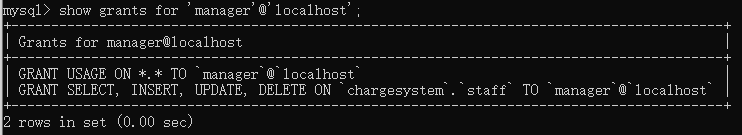
1. [数据库的使用](#_数据库的使用)
2. [hosue\_owner](#house_owner)
3. [house](#house)
4. [department](#department)
5. [staff](#dbstaff)
6. [db库](#_db库)
7. [DBINFO](#DBINFO)
8. [SQLString](#SQLString)
9. [insert](#insert)
10. [delete](#delete)
11. [select](#select)
12. [update](#update)
13. [staff库](#_staff库)
14. [Staff](#Staff)
15. [Manager](#Manager)
16. [chargeStaff](#chargeStaff)
17. [cs库](#_cs库)
18. [chargeDisplay库](#_chargeDisplay库)
19. [managerDisplay库](#_managerDisplay库)

# 数据库的使用

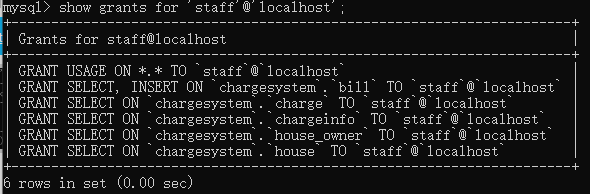
在项目文件夹下的database文件夹下存放着导出数据库文件cs.sql。根据和数据库文件一起存放的readme.txt导入数据库。

对此数据库存在2名用户： manager和staff（需要在导入数据库后自行创建这两个用户）

manager的权限



staff的权限



该数据库共包含4个表house\_owner、house、department和staff

1. house\_owner——业主信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 数据类型 | 属性 | 描述 |
| owner\_id | varchar | 主键 | 业主号 |
| owner\_name | varchar | - | 业主姓名 |
| work\_place | varchar | - | 业主工作地址 |
| phone\_number | varchar | - | 业主手机号码 |

1. house——房屋信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 数据类型 | 属性 | 描述 |
| house\_id | varchar | 主键 | 房屋号 |
| area | double | - | 房屋面积 |
| room | double | - | 房屋的房间数量 |
| owner\_id | varchar(20) | 外键 | 房主ID |

1. department——部门信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 数据类型 | 属性 | 描述 |
| department\_id | varchar | 主键 | 部门号 |
| department\_name | varchar | - | 部门名称 |
| manager | varchar | - | 部门负责人 |
| phone\_number | varchar | - | 部门电话 |

1. staff——员工信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 数据类型 | 属性 | 描述 |
| staff\_id | varchar | 主键 | 员工号 |
| staff\_name | varchar | - | 员工姓名 |
| birthday | date | - | 生日 |
| gender | enum(‘M’,’F’) | - | 性别，M为男，F为女 |
| address | varchar | - | 员工住址 |
| phone | varchar | - | 员工电话 |
| department\_id | varchar | 外键 | 员工所属部门号 |
| job | enum(‘M’,’S’) | - | 职位，M为经理，S为员工 |
| pwd | varchar | - | 员工密码 |

1. bill——账单表

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 数据类型 | 属性 | 描述 |
| property | varchar(20) | - | 物业费 |
| clean | varchar(20) | - | 卫生费 |
| water | varchar(20) | - | 水费 |
| electricity | varchar(20) | - | 电费 |
| time\_stamp | date | - | 添加此条记录的时间 |
| house\_id | varchar | - | 房屋号 |
| state | ENUM('unordered',"unpaid") | - | 状态，未出单、未支付 |
| staff\_id | varchar(20) | - | 插入此条收费记录的员工的ID |

# db库

db库主要包含于数据库相关的操作类

**DNINFO：**

此类无方法。该类为delete、insert、select、update类提供支持数据支持。

**SQLString：**

该类负责为delete、insert、select、update类构造SQL语句

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 返回类型 | 参数 | 描述 |
| construct | String | (String[] content) | 接受一个数组，将其拼接后返回 |
| insertColumn | String | (String[] centent) | 接受一个数组，将其拼接为insert语句要注入的字段 |
| insertValues | String | (String[] content) | 接受一个数组，将其拼接为insert语句要注入的内容 |

**insert：**

该类负责执行SQL语句中的insert语句

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 返回类型 | 参数 | 描述 |
| insertSet | void | (String user, String pass, String table, String[] column, String[] content) | user、pass为执行此操作的对象的用户名和密码，table是要插入的表的名字，column是要插入的字段的集合，content是要插入的内容 |

**delete：**

该类负责执行SQL语句中的delete语句

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 返回类型 | 参数 | 描述 |
| deleteSet | void | (String user, String pass, String table, String column, String sign) | user、pass为执行此操作的对象的用户名和密码，table是执行操作的表的名字，column是字段名，sign是要删除的行对应的column的值 |

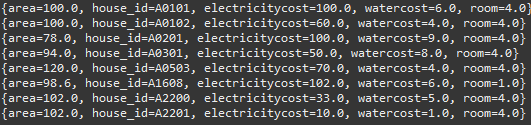
**select：**

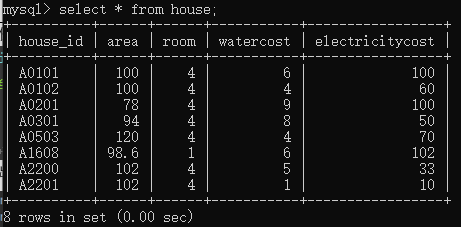
该类负责执行SQL语句中的select语句

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 返回类型 | 参数 | 描述 |
| selectSet | Object | (String user, String pass, String table, String content, String[] selectInfo) | user、pass为执行此操作的对象的用户名和密码，table是要选择的表的名字，content是要选择的字段，selectInfo是附加信息，用于构造where部分语句，其内容为一个包含两个字符串的数组，第一个字符串表示字段名，第二个字符串表示该字段的值。不需要限定查找字段时传入null |
| selectSet | List<HashMap<String,Object>> | (String user, String pass, String table, String[] content, String[] selectInfo) | user、pass为执行此操作的对象的用户名和密码，table是要选择的表的名字，content是要选择的字段构成的数组，selectInfo是附加信息，用于构造where部分语句，其内容为一个包含两个字符串的数组，第一个字符串表示字段名，第二个字符串表示该字段的值。不需要selectInfo时传入null。  由select得到的每一行信息构成一个HashMap储存在List中。  HashMap中的key是字段名，value是字段对应的值 |

用例：

|  |
| --- |
| List<HashMap<String,Object>> recieve=(List<HashMap<String, Object>>) *selectSet*("staff","staff","house","\*",null);  for(HashMap<String,Object> m:recieve) {  System.***out***.println(m);  } |

 输出：

 对比数据库：

**update：**

该类负责执行SQL语句中的delete语句

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 返回类型 | 参数 | 描述 |
| uppdateSet | void | (String user, String pass, String table, String[] column, String content[], String[] sign) | user、pass为执行此操作的对象的用户名和密码，table是要跟新的表的名字，content是要选择的字段构成的数组，sign是附加信息，用于构造where部分语句，其内容为一个包含两个字符串的数组，第一个字符串表示字段名，第二个字符串表示该字段的值。 |

# staff库

staff库主要包含和员工相关的库

**Staff：**

员工类，基类，本身没什么用

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 返回类型 | 参数 | 描述 |
| Staff | - | (String staffID) | staff类的构造函数，接受一个String字符串员工号 |
| getStaffID | String | - | 返回此对象的员工号 |

**Manager：**

经理类，Staff类的子类

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 返回类型 | 参数 | 描述 |
| Manager | - | (String staffID) | Manager的构造函数，接受一个String字符串员工号 |
| getStaffID | String | - | 返回员工号 |
| getUser | String | - | 返回Manager的数据库账号 |
| getPass | String | - | 返回Manager的数据库密码 |

**chargeStaff：**

收费员工类，Staff类的子类

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 返回类型 | 参数 | 描述 |
| charge | double[] | (String houseID) | houseID表示要收费的房屋的房屋号，返回一个收取费用的数组。数组包含物业费和卫生费 |
| houseInfo | double[] | (String houseID) | 返回房屋的面积和房间数量信息 |
| getID | String | - | 返回员工号 |
| getUser | String | - | 返回chargeStaff的数据库账号 |
| getPass | String | - | 返回chargeStaff的数据库密码 |

# cs库

本库是整个程序的主显示界面库。cs是ChargeSystem的缩写。

**DisPlay：**

整个程序的主显示界面



**logIn：**

进行登录判断的模块

**failedToLogin：**

登录失败后显示的界面



**judgingPosition：**

判断登录员工的职位的模块

# chargeDisplay库

本库是收费员工的显示界面

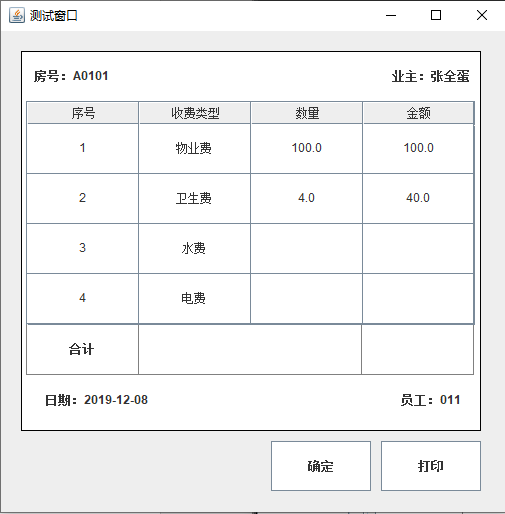
**Main：**

收费员工的主显示界面



**charge：**

收费界面



**ChargeSuccess：**

收费成功后的提示界面



**myButton：**

用于方便的管理自定义的按钮样式，内部包含两个方法，返回不同风格的按钮

**myLabel：**

用于方便的管理自定义的标签样式，内部包含两个方法，返回不同风格的标签

**myTable：**

用于方便的管理自定义的表格样式，内部包含两个方法，返回不同风格的表格

**reservedDigits：**

用于返回保留一位小数的小数

**toChinese：**

将数字字符串转换为中文大写的字符串

# managerDisplay库

尚待施工