

编 号

江南大学

# 数据库系统课程设计

题目： 汽车租赁管理子系统

学 院 人工智能与计算机

专 业 计算机科学与技术

学 号 1033190609

学生姓名 刘鸿嘉

任课教师 钱瑛

二〇一九年六月

# 设计总说明

随着信息技术的快速发展,信息化正改变着人们的生活与交易方式,越来越多的传统业务开始使用信息系统替代原有的交易模式,汽车租赁行业便是其中之一.面对飞速发展的汽车租赁市场,传统人工管理的方式在进行繁项的业务信息处理时显得极为不适应,因为工作和生活的缘故,在必要的时候向汽车出租公司提出租赁车辆的服务要求,已成为大众广为认可的选择,在这种社会需求的带动下,汽车租赁市场出现了前所未有的一片繁荣景象,各个汽车租赁公司门庭若市,业务激增,然而在繁忙的景象背后,却也暴露出汽车租赁公司在处理业务信息时以人工方式所带来的效率低下、管理混乱、失误增多等诸多弊端,在实践中,汽车租赁行业的业主们逐渐意识到在租赁行业中借助计算机引入信息管理的重要性,并开始不断提高自身的素质和高科技在工作中的投入,以改善信息的管理,提高服务效率,汽车租赁行业庞大,竞争激烈,通过信息化方式如网站、微信号等推广自身品牌,实现更为便捷的浏览与支付的交易渠道,都能更有效地扩张与占领市场,面向多终端的交易与管理系统又是最新的发展趋势,随着手机等移动设备功能的增强,越来越多的设计、操作与交互在手机终端上可以得到实现,因此,充分利用网络信息渠道与智能终端,构建出完整的汽车租赁业务的查阅浏览、交易、订单管理的面向多平台的汽车租赁管理系统是实现高效业务的必要手段,随着新形势下的互联网出租车模式如滴滴等的侵入,传统汽车租赁行业更应跟上时代潮流,开发出属于自己的运营系统。

前端主要应用 Vue.js,Vue.js 是一套构建用户界面的渐进式框架。与其他重量级框架不同的是,Vue 采用自底向上增量开发的设计。Vue 的核心库只关注视图层,并且易于学习,非常容易与其它库或已有项目整合。另一方面,Vue 完全有能力驱动采用单文件组件和 Vue 生态系统支持的库开发的复杂单页应用。

后端采用 python 语言进行设计, Spyder 是 Python 专用的一种开源 IDE,其独特之处在于专为数据科学工作流程进行了优化。它与 Anaconda 软件包管理器捆绑在一起,后者是 Python 编程语言的标准发行版。Spyder 拥有所有必需的 IDE 特性,包括代码完整性及集成文件浏览器。

数据库采用 MySQL,MySQL 是一个关系型数据库管理系统,由瑞典 MySQL AB 公司开发,属于 Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统之一,在 WEB 应用方面,MySQL 是最好的 RDBMS(Relational Database Management System,关系数据库管理系统)应用软件之一。

本系统分为 8 个模块:登录模块,登录账号信息修改模块,用户信息管理模块,客户信息管理模块,司机信息管理模块,车辆信息管理模块,驾驶信息管理模块,订单信息管理模块。

**关键词:** 汽车租赁; 管理; Python; MySQL ; Vue

# 目录

设计总说明 .....	2
目录 .....	3
第一章 绪论 .....	5
1.1 研究的目的及意义 .....	5
第二章 需求分析 .....	5
2.1 可行性分析 .....	5
第三章 系统设计 .....	6
3.1 系统设计概述 .....	6
3.1.1 设计思路 .....	6
3.1.2 设计目标 .....	6
3.2 系统功能模块设计 .....	6
3.2.1 登录模块功能划分 .....	7
3.2.2 登录账号信息修改模块功能划分 .....	8
3.2.3 客户信息管理模块功能划分 .....	8
3.2.4 司机信息管理模块功能划分 .....	8
3.2.5 车辆信息管理模块功能划分 .....	9
3.2.6 驾驶信息管理模块功能划分 .....	10
3.2.7 订单信息管理模块功能划分 .....	10
3.2.8 用户信息管理模块功能划分 .....	11
3.3 系统数据库设计 .....	12
3.3.1 E-R 图 .....	12
3.3.2 数据库 .....	14
第 4 章 系统的实现与维护 .....	15
4.1 系统的实现 .....	15
4.1.1 开始界面 .....	15
4.1.2 登录界面 .....	16
4.1.3 登录账号信息修改模块界面 .....	17
4.1.4 用户信息管理模块界面 .....	18
4.1.5 客户信息管理模块界面 .....	20
4.1.6 司机信息管理模块界面 .....	23
4.1.7 车辆信息管理模块界面 .....	25
4.1.8 驾驶信息管理模块界面 .....	28
4.1.9 订单信息管理模块界面 .....	29
第五章 运行环境配置 .....	34
5.1 后端环境 .....	34
5.2 Vue 搭建 .....	35
5.3 MySQL 搭建 .....	35
第六章 后端操作 .....	36
6.1 cars.py .....	36

第七章 前端操作.....	58
7.1 程序结构 .....	58
7.2 main.js.....	59
7.3 login.vue.....	59
7.4 Index.vue.....	63
7.5 App.vue.....	64
7.6 Index.js .....	65
7.7 vehicle_admin.vue .....	66
7.8 rent_order_admin.vue.....	76
7.9 driver_admin.vue.....	102
7.10 drive_admin.vue.....	111
7.11 customer_admin.vue .....	119
7.12 account_admin.vue .....	128
7.13 Sidebar.vue .....	138
7.14 Home.vue .....	140
7.15 Header.vue .....	141
MySQL 数据库操作.....	144

# 第一章 绪论

## 1.1 研究的目的及意义

随着信息技术的快速发展,信息化正改变着人们的生活与交易方式,越来越多的传统业务开始使用信息系统替代原有的交易模式,汽车租赁行业便是其中之一.面对飞速发展的汽车租赁市场,传统人工管理的方式在进行繁项的业务信息处理时显得极为不适应,因为工作和生活的缘故,在必要的时候向汽车出租公司提出租赁车辆的服务要求,已成为大众广为认可的选择,在这种社会需求的带动下,汽车租赁市场出现了前所未有的一片繁荣景象,各个汽车租赁公司门庭若市,业务激增,然而在繁忙的景象背后,却也暴露出汽车租赁公司在处理业务信息时以人工方式所带来的效率低下、管理混乱、失误增多等诸多弊端,在实践中,汽车租赁行业的业主们逐渐意识到在租赁行业中借助计算机引入信息管理的重要性,并开始不断提高自身的素质和高科技在工作中的投入,以改善信息的管理,提高服务效率,汽车租赁行业庞大,竞争激烈,通过信息化方式如网站、微信号等推广自身品牌,实现更为便捷的浏览与支付的交易渠道,都能更有效地扩张与占领市场,面向多终端的交易与管理系统又是最新的发展趋势,随着手机等移动设备功能的增强,越来越多的设计、操作与交互在手机终端上可以得到实现,因此,充分利用网络信息渠道与智能终端,构建出完整的汽车租赁业务的查阅浏览、交易、订单管理的面向多平台的汽车租赁管理系统是实现高效业务的必要手段,随着新形势下的互联网出租车模式如滴滴等的侵入,传统汽车租赁行业更应跟上时代潮流,开发出属于自己的运营系统。

## 第二章 需求分析

### 2.1 可行性分析

前端主要应用 Vue.js,Vue.js 是一套构建用户界面的渐进式框架。与其他重量级框架不同的是,Vue 采用自底向上增量开发的设计。Vue 的核心库只关注视图层,并且易于学习,非常容易与其它库或已有项目整合。另一方面,Vue 完全有能力驱动采用单文件组件和 Vue 生态系统支持的库开发的复杂单页应用。

后端采用 python 语言进行设计,Spyder 是 Python 专用的一种开源 IDE,其独特之处在于专为数据科学工作流程进行了优化。它与 Anaconda 软件包管理器捆绑在一起,后者是 Python 编程语言的标准发行版。Spyder 拥有所有必需的 IDE 特性,包括代码完整性及集成文件浏览器。

数据库采用 MySQL,MySQL 是一个关系型数据库管理系统,由瑞典 MySQL AB 公司开发,属于 Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统之一,在 WEB 应用方面,MySQL 是最好的 RDBMS(Relational Database Management System,关系数据库管理系统)应用软件之一。

以上均有工程实际应用前例,且逻辑完备,因此具有一定可行性。

## 第三章 系统设计

### 3.1 系统设计概述

#### 3.1.1 设计思路

前端构建网页框架,通过 post 接口与后端相连,向后端传输用户输入信息,并获取后端信息在页面进行显示。后端通过 post 接口与前端进行连接,获取前端数据,并通过相应的 sql 语句对数据库进行相关操作,获取数据库信息传至前端。

#### 3.1.2 设计目标

车辆租赁部门希望借助计算机对车辆租赁使用情况进行全面管理。

- (1) 基础信息设置: 包括车辆类型、车辆名称、租赁模式等的设置。
- (2) 信息管理: 车辆信息、司机信息、客户信息的维护和查询。
- (3) 综合业务管理: 对综合业务进行管理和查询,包括汽车预约、出租、续租、催车、还车结算等。
- (4) 租金统计: 可以进行当天租金统计和一定时间段的租金统计。
- (5) 系统管理: 包括系统维护、权限设置、更改密码等。

### 3.2 系统功能模块设计

本系统分为 8 个模块: 登录模块, 登录账号信息修改模块, 用户信息管理模块, 客户信息管理模块, 司机信息管理模块, 车辆信息管理模块, 驾驶信息管理模块, 订单信息管理模块。

以下为功能模块划分:

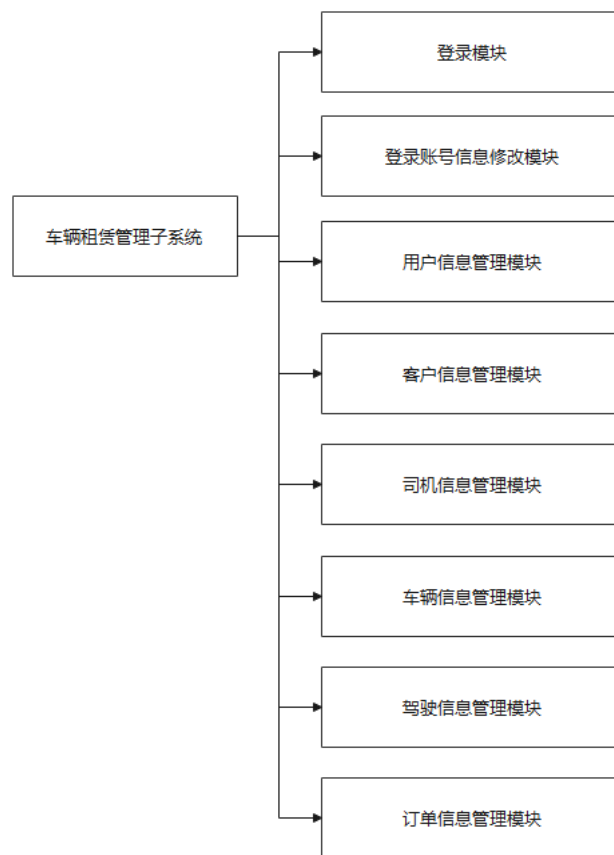


图 3-1 整体功能图

### 3.2.1 登录模块功能划分

登录模块可以实现用户和管理员的登录操作, 功能划分如图:

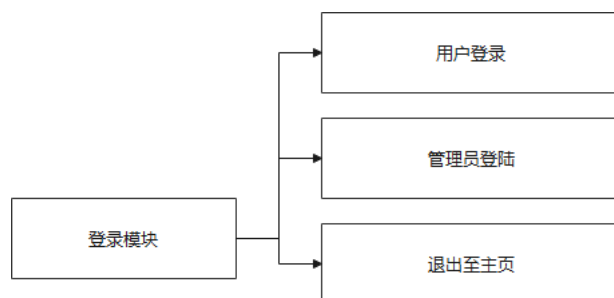


图 3-2 登录模块功能图

### 3.2.2 登录账号信息修改模块功能划分

用户借阅模块可以实现用户查询图书信息和借阅图书操作, 功能划分如图:

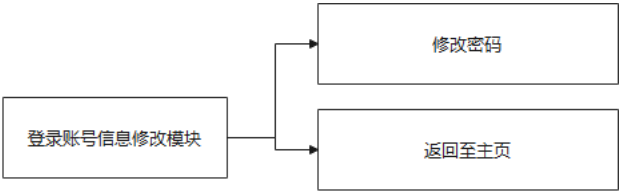


图 3-3 用户借阅模块功能图

### 3.2.3 客户信息管理模块功能划分

客户信息管理模块可以实现添加客户信息、添加客户信息、返回主页、修改客户信息, 删除客户信息

功能划分如图:

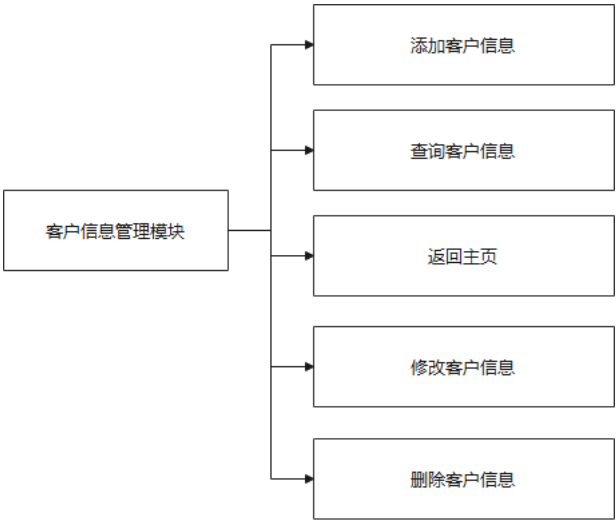


图 3-4 客户信息管理模块功能图

### 3.2.4 司机信息管理模块功能划分

司机信息管理模块可以实现用户添加司机信息, 查询司机信息, 返回主页, 修改司机信息, 删除司机信息

功能划分如图:



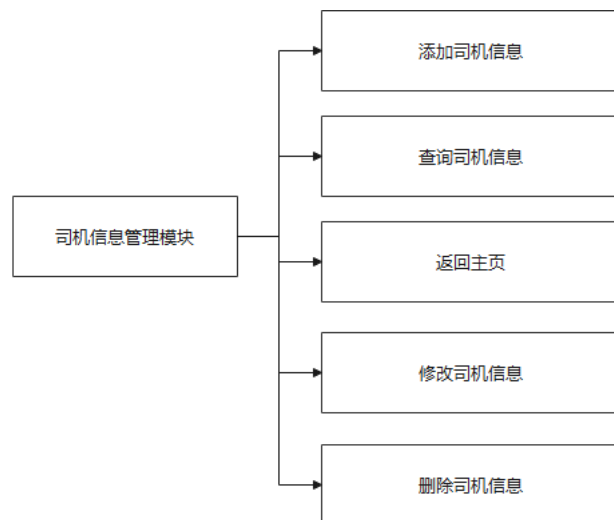


图 3-5 司机信息管理模功能图

### 3.2.5 车辆信息管理模块功能划分

车辆信息管理模块可以实现添加车辆信息、查询车辆信息、查询车辆信息、查询已出租车辆, 返回主页, 更新目录, 修改车辆信息, 删除车辆信息  
功能划分如图:

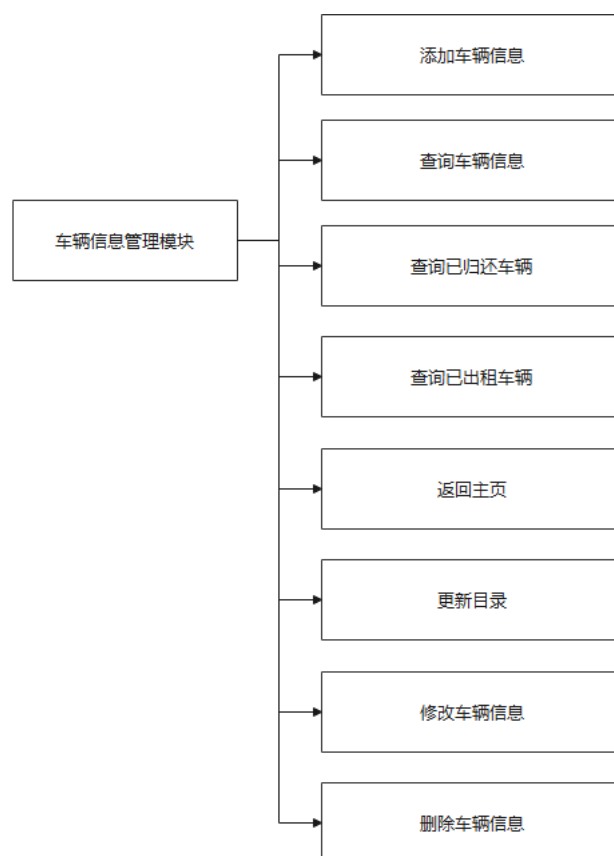


图 3-6 车辆信息管理模块功能图

### 3.2.6 驾驶信息管理模块功能划分

驾驶信息管理模块可以实现驾驶添加司机信息, 查询驾驶信息, 返回主页, 修改驾驶信息, 删除驾驶信息

功能划分如图:

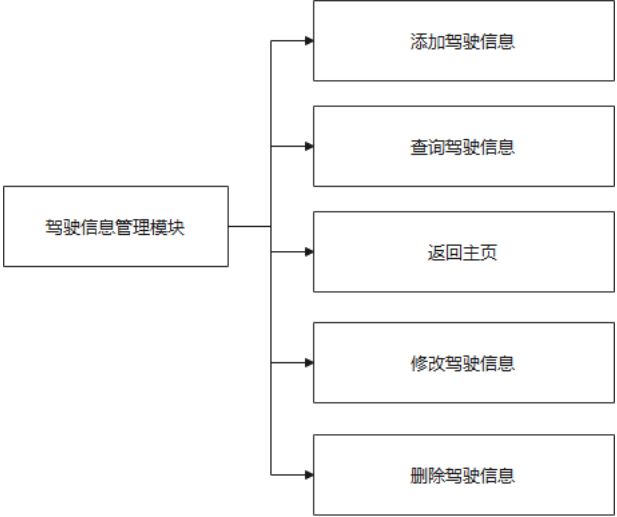


图 3-7 用户信息管理模块功能图

### 3.2.7 订单信息管理模块功能划分

订单信息管理模块可以实现添加订单信息、实付款统计、押金统计、查询订单信息, 返回主页, 修改订单信息, 修改订单信息, 查看订单客户信息, 查看订单司机信息, 查看订单车辆信息

功能划分如图:

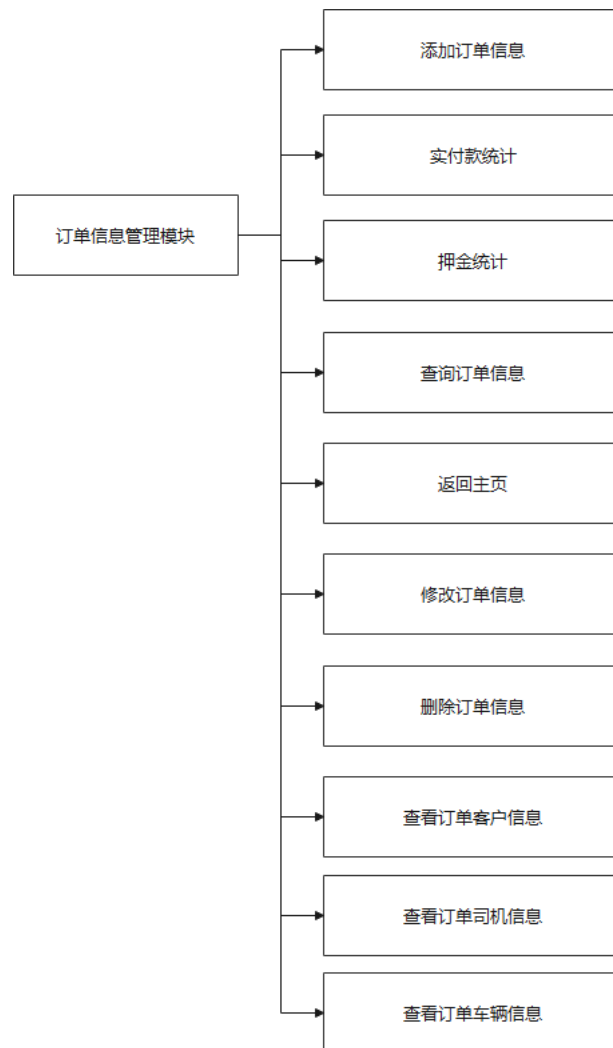


图 3-8 订单信息管理模块功能图

### 3.2.8 用户信息管理模块功能划分

用户信息管理模块可以实现添加账号, 修改账号信息, 删除账号信息功能划分如图:

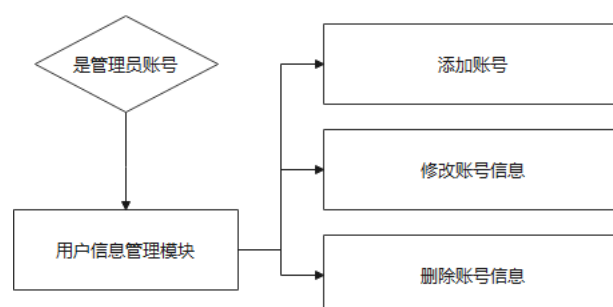
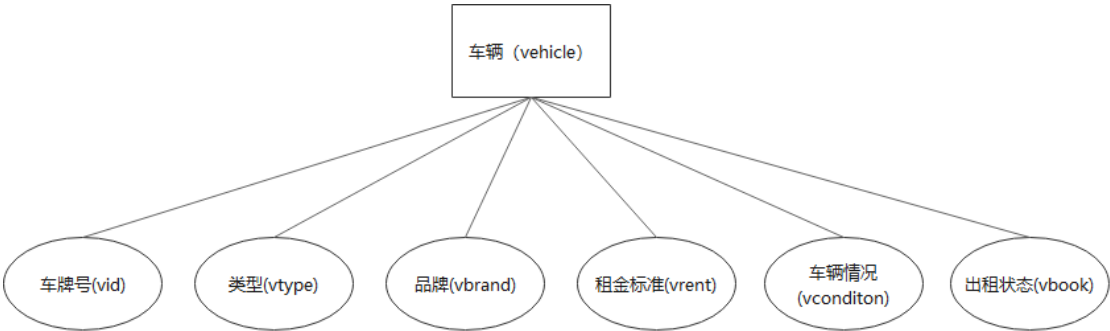


图 3-9 用户信息管理模块功能图

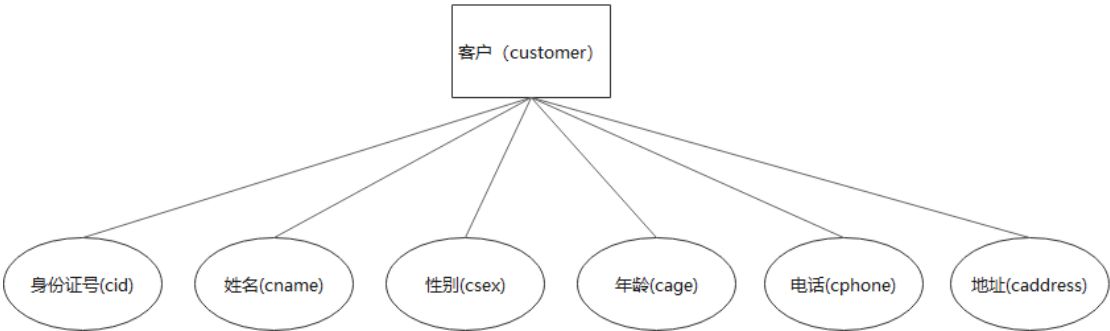
### 3.3 系统数据库设计

#### 3.3.1 E-R 图

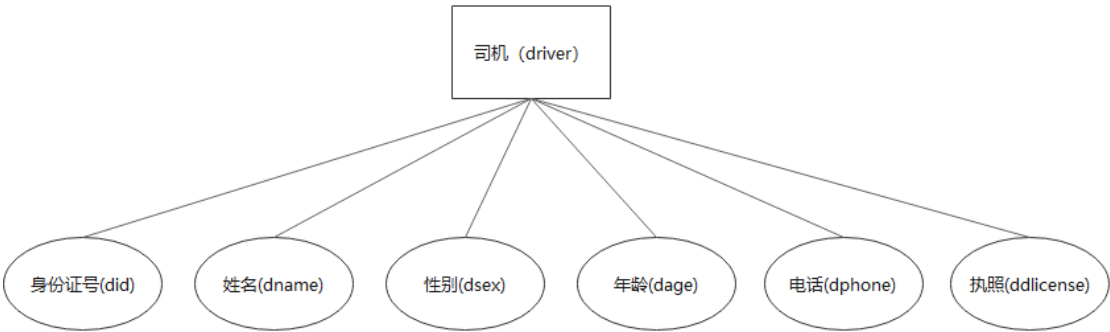
##### 3.3.1.1 车辆实体：



##### 3.3.1.2 客户实体：



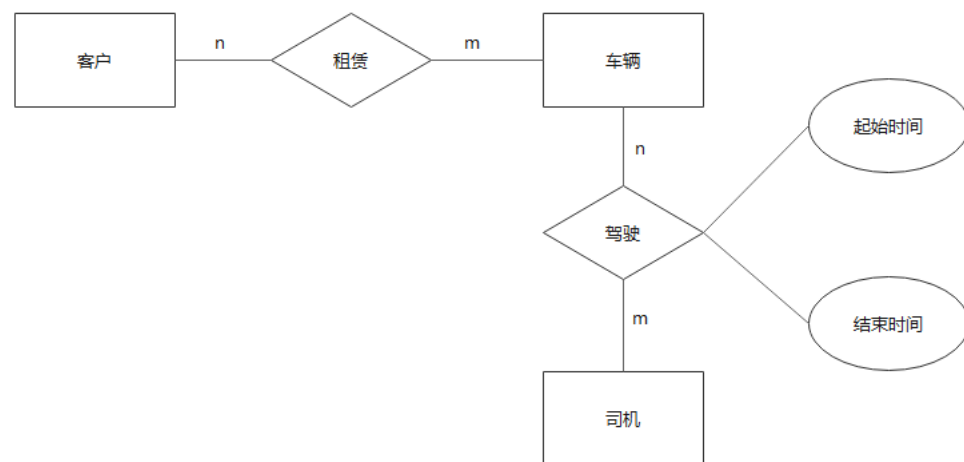
##### 3.3.1.3 司机实体：



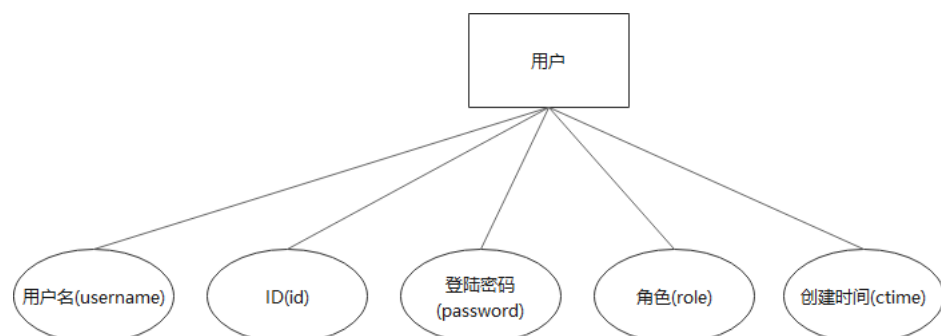
#### 3.3.1.4 订单实体:



#### 3.3.1.5 关系图:




#### 3.3.1.6 用户实体:



## 3.3.2 数据库


### 3.3.2.1 customer 表:

名	类型	长度	小数点	不是 null	虚拟	键
cid	char	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
cname	varchar	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
csex	varchar	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
cage	int			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
cphone	varchar	50		<input type="checkbox"/>	<input type="checkbox"/>	
caddress	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	



### 3.3.2.2 vehicle 表:

名	类型	长度	小数点	不是 null	虚拟	键
vid	varchar	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
vtype	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
vbrand	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
vrent	float			<input type="checkbox"/>	<input type="checkbox"/>	
vcondition	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
vdel	int			<input type="checkbox"/>	<input type="checkbox"/>	
vbook	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	

### 3.3.2.3 driver 表:

名	类型	长度	小数点	不是 null	虚拟	键
did	char	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
dname	varchar	50		<input type="checkbox"/>	<input type="checkbox"/>	
dsex	varchar	50		<input type="checkbox"/>	<input type="checkbox"/>	
dage	int			<input type="checkbox"/>	<input type="checkbox"/>	
dphone	varchar	50		<input type="checkbox"/>	<input type="checkbox"/>	
ddlicense	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	

### 3.3.2.4 drive 表:

名	类型	长度	小数点	不是 null	虚拟	键
vid	varchar	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
did	char	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	 2
startdate	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
enddate	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	

外键:

名	字段	被引用的模式	被引用的表 (父)	被引用的字段	删除时	更新时
drive_ibfk_1	did	data_1	driver	did	RESTRICT	RESTRICT
drive_ibfk_2	vid	data_1	vehicle	vid	RESTRICT	RESTRICT


### 3.3.2.5 rent\_order 表:

字段	索引	外键	触发器	选项	注释	SQL 预览					
名		字段				类型	长度	小数点	不是 null	虚拟	键
vid						varchar	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
cid						char	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
startdate						varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
enddate						varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
model						varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
adv_pay						int			<input type="checkbox"/>	<input type="checkbox"/>	
act_pay						int			<input type="checkbox"/>	<input type="checkbox"/>	
did						char	50		<input type="checkbox"/>	<input type="checkbox"/>	
rid						int	10		<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
bre_pay						int			<input type="checkbox"/>	<input type="checkbox"/>	
state						varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
act_enddate						varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	

外键:

字段	索引	外键	触发器	选项	注释	SQL 预览					
名		字段			被引用的模式	被引用的表 (父)	被引用的字段	删除时	更新时		
cid_ord		cid			data_1	customer	cid	RESTRICT	RESTRICT		
did_ord		did			data_1	driver	did	RESTRICT	RESTRICT		
vid_ord		vid			data_1	vehicle	vid	RESTRICT	RESTRICT		

### 3.3.2.6 login 表:

字段	索引	外键	触发器	选项	注释	SQL 预览					
名						类型	长度	小数点	不是 null	虚拟	键
id						int			<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
username						varchar	100		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
password						varchar	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
role						tinyint			<input type="checkbox"/>	<input type="checkbox"/>	
ctime						datetime			<input type="checkbox"/>	<input type="checkbox"/>	

## 第 4 章 系统的实现与维护

### 4.1 系统的实现

#### 4.1.1 开始界面

点击”请点击此处进入”,进入登录界面。



#### 4. 1. 2 登录界面

\* 用户名

请输入用户名

\* 密码

请输入密码

登录

返回

\* 用户名

请输入用户名

请输入用户名

\* 密码

请输入密码

请输入密码

登录

返回



\* 用户名

\* 密码

登录

返回

输入错误:

✖ 用户名或密码错误

登录成功:

✔ 登录成功

#### 4. 1. 3 登录账号信息修改模块界面



修改密码

退出

×

修改密码

新密码

请输入新密码

确认密码

请输入确认密码

取消

确认

#### 4.1.4 用户信息管理模块界面

车辆租赁管理系统

admin

用户管理

车辆管理

客户管理

司机管理

驾驶信息管理

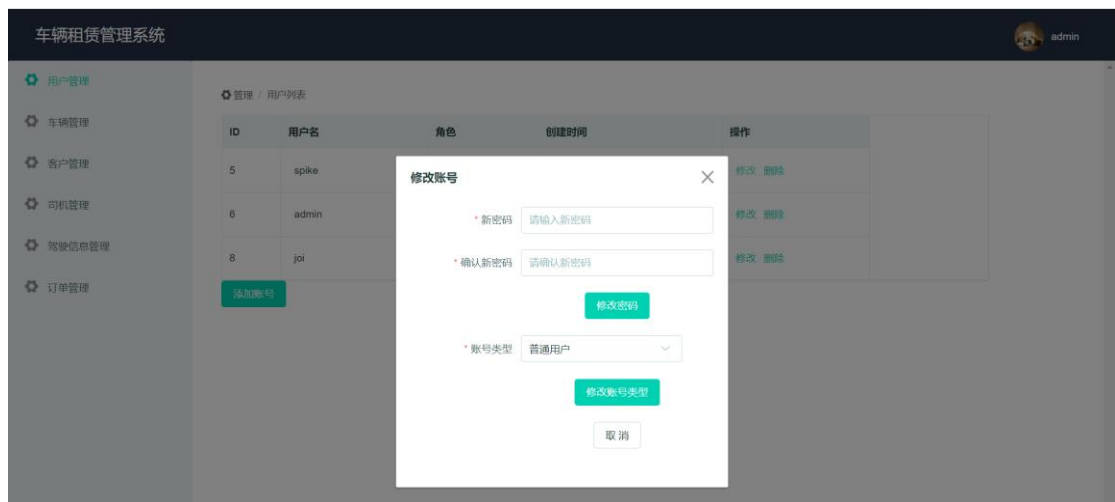
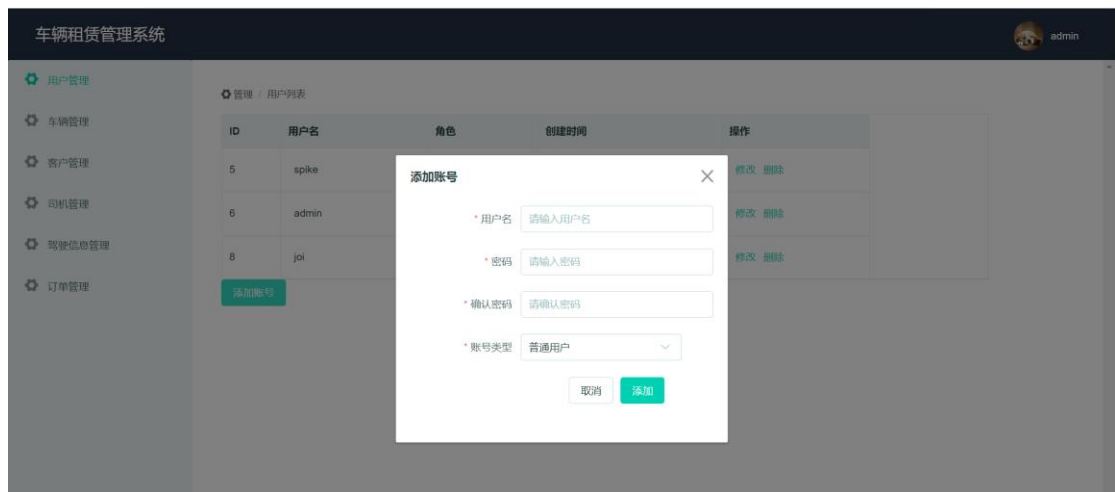
订单管理

管理 / 用户列表

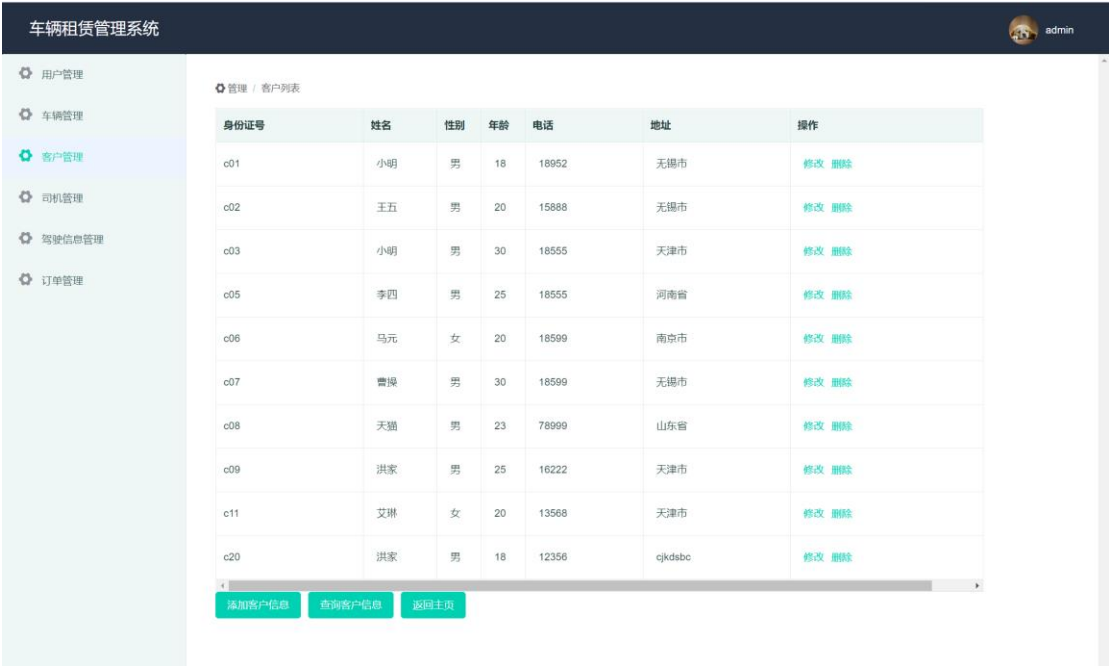
ID	用户名	角色	创建时间	操作
5	spike	普通用户	Thu, 08 Jul 2021 16:02:55 GMT	修改 删除
6	admin	管理员	Fri, 09 Jul 2021 13:09:36 GMT	修改 删除
8	joi	管理员	Mon, 12 Jul 2021 10:39:05 GMT	修改 删除

添加账号

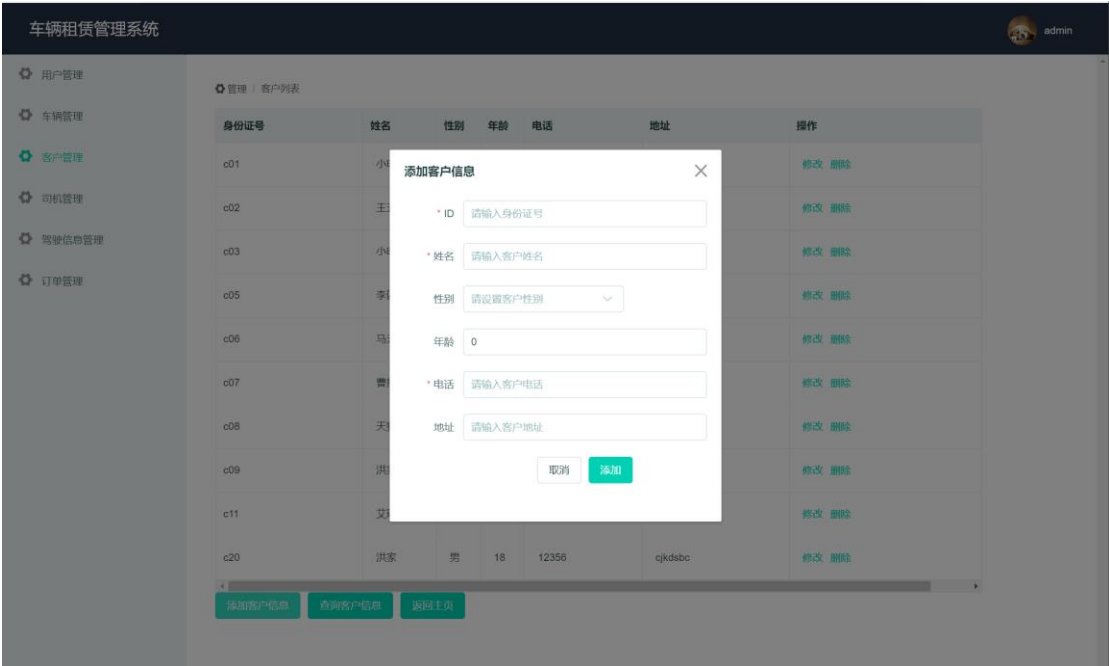
添加账号:

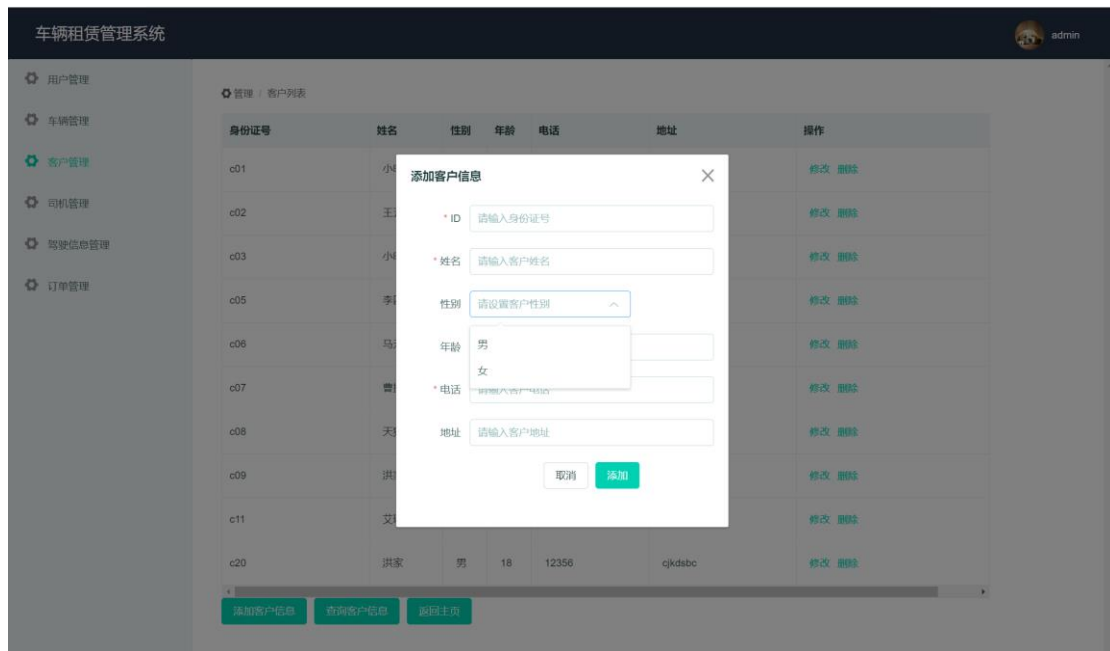


## 4.1.5 客户信息管理模块界面

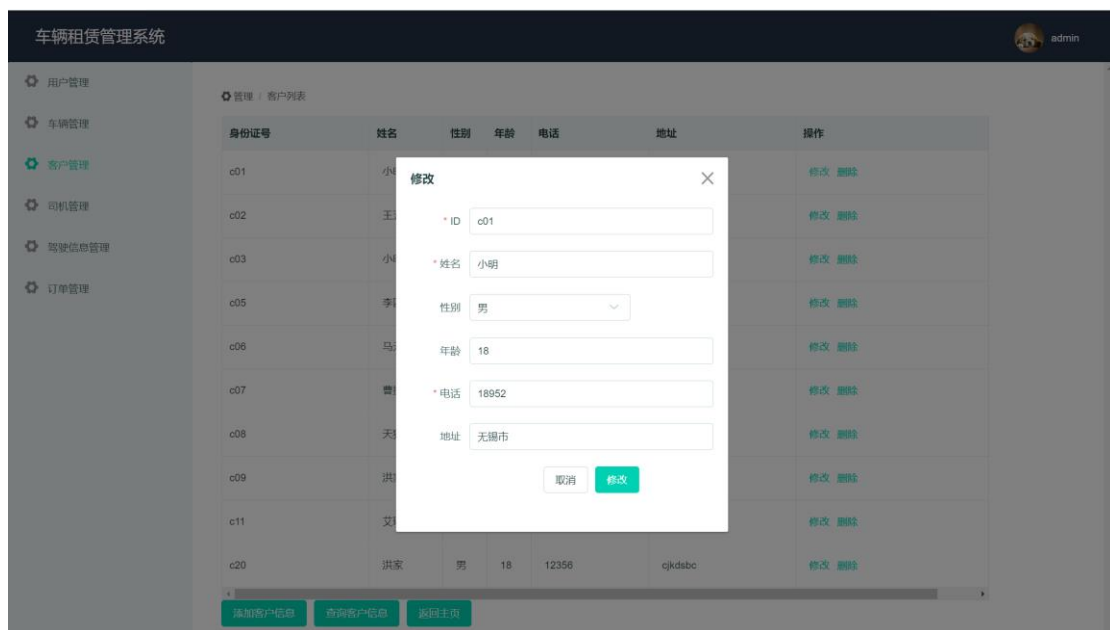


添加客户信息:





修改客户信息:



查询客户信息:

车辆租赁管理系统

admin

用户管理

车辆管理

客户管理

司机管理

驾驶信息管理

订单管理

管理 / 客户列表

身份证号	姓名	性别	年龄	电话	地址	操作
c01	小明	男	18	18952	无锡市	修改 删除
c02	王					修改 删除
c03	小					修改 删除
c05	李					修改 删除
c06	马元	女	20	18599	南京市	修改 删除
c07	曹瑞	男	30	18599	无锡市	修改 删除
c08	天强	男	23	78999	山东省	修改 删除
c09	洪家	男	25	16222	天津市	修改 删除
c11	艾琳	女	20	13568	天津市	修改 删除
c20	洪家	男	18	12356	qkdsbc	修改 删除

添加客户信息 查询客户信息 返回主页

车辆租赁管理系统

admin

用户管理

车辆管理

客户管理

司机管理

驾驶信息管理

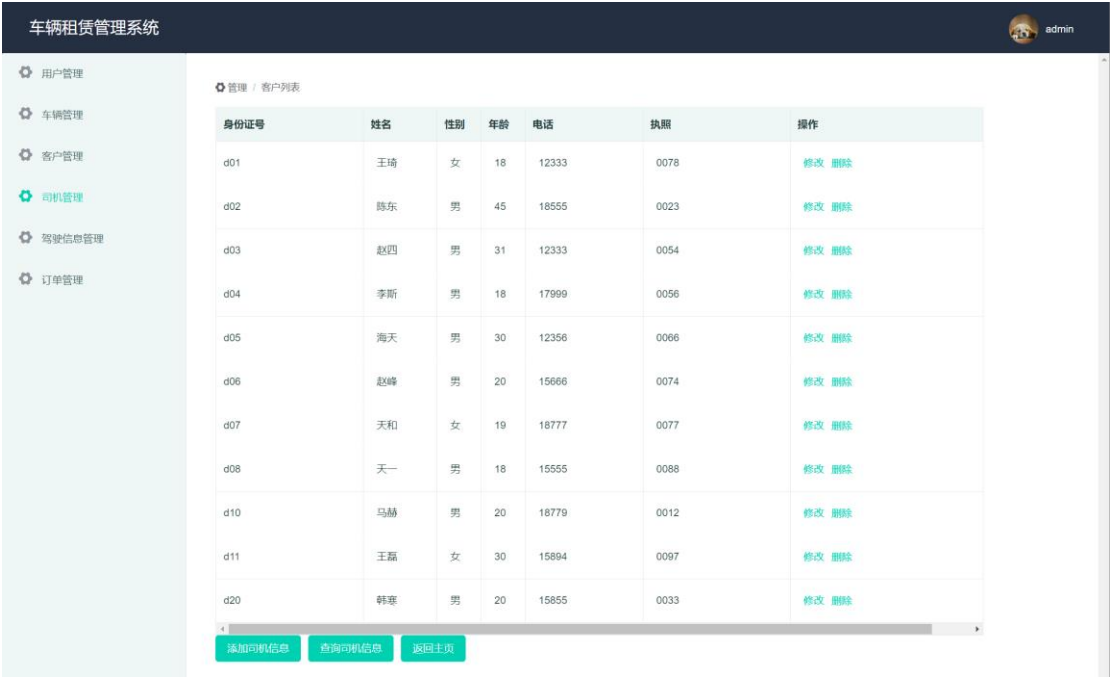
订单管理

管理 / 客户列表

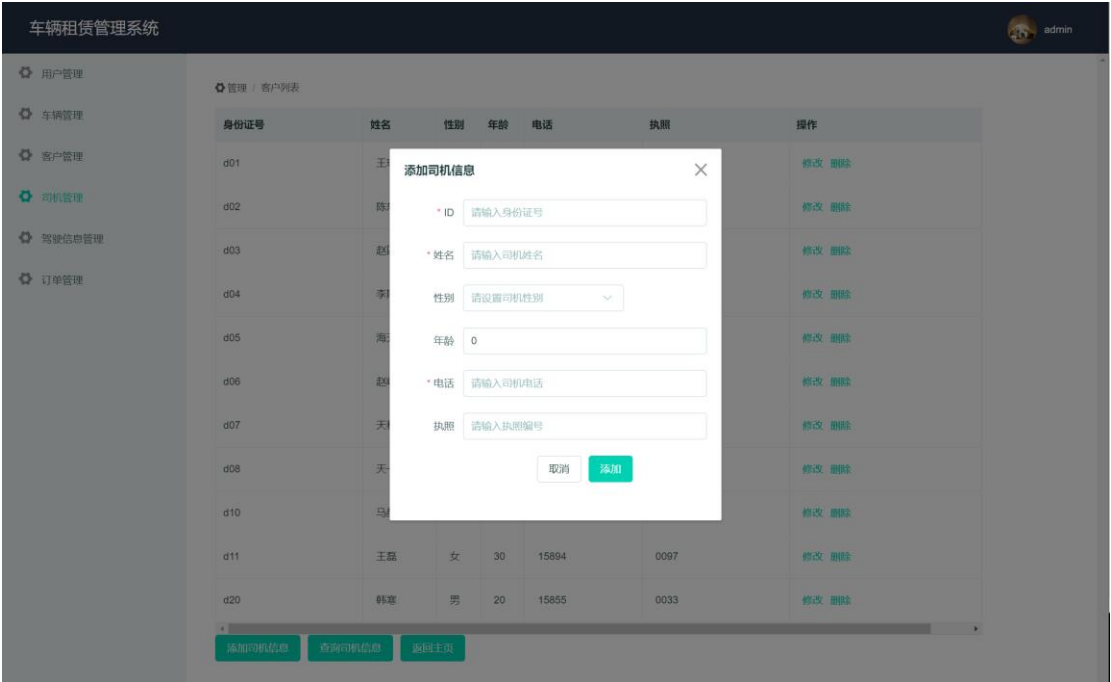
身份证号	姓名	性别	年龄	电话	地址	操作
c01	小明	男	18	18952	无锡市	修改 删除

添加客户信息 查询客户信息 返回主页

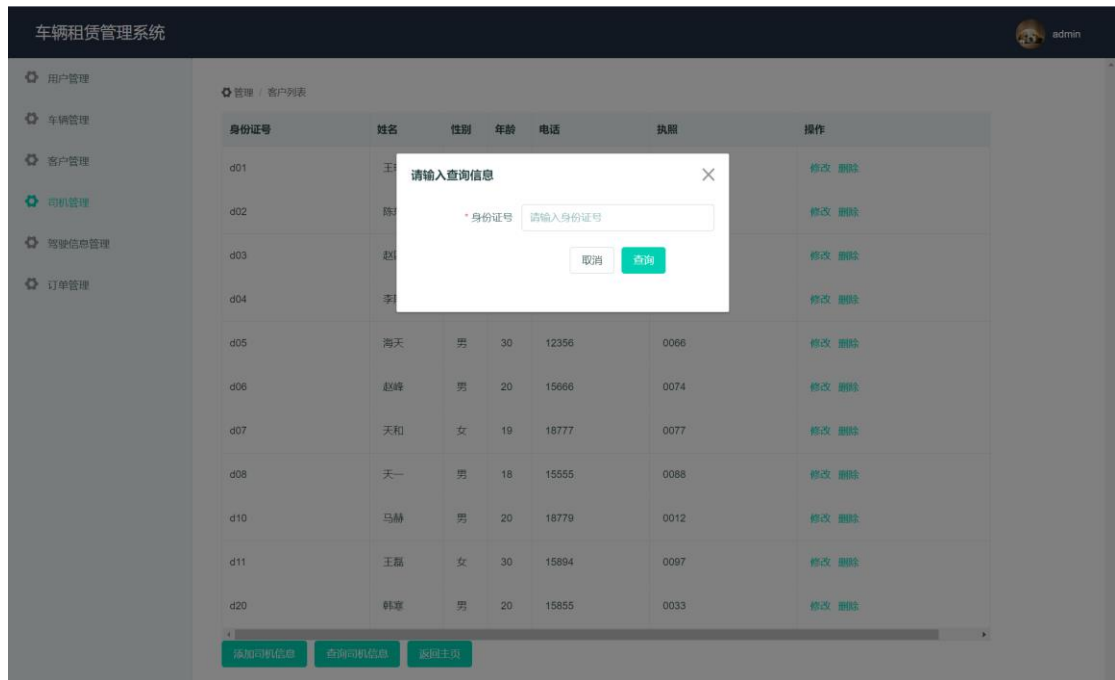
## 4.1.6 司机信息模块界面



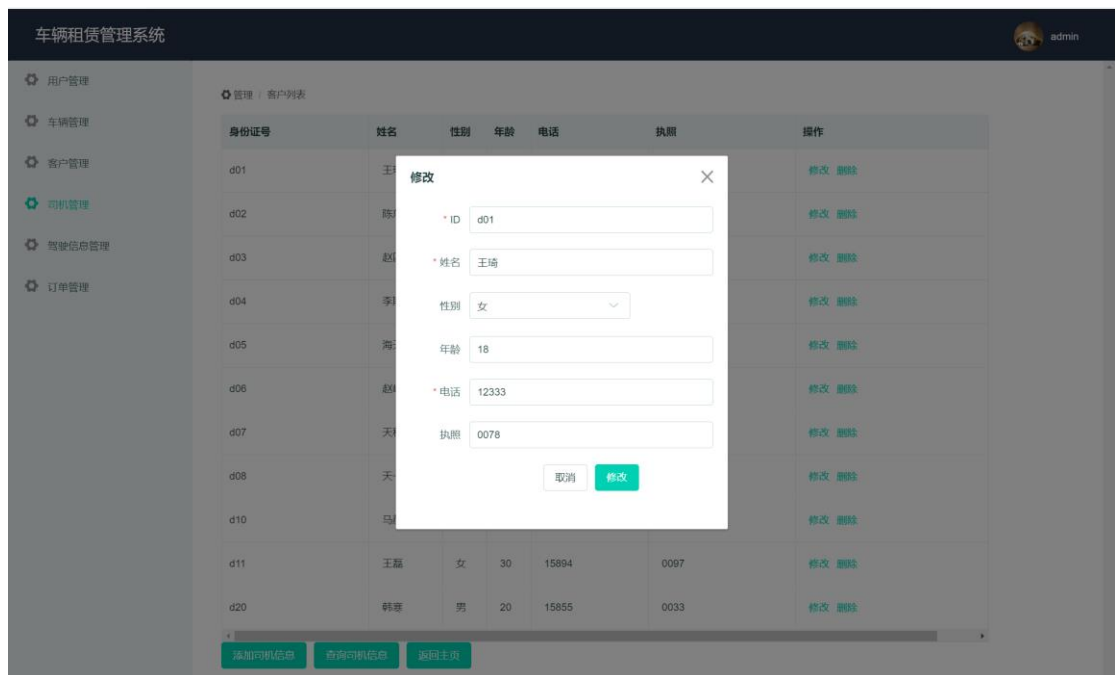
添加司机信息:



查询司机信息:



修改车辆信息：

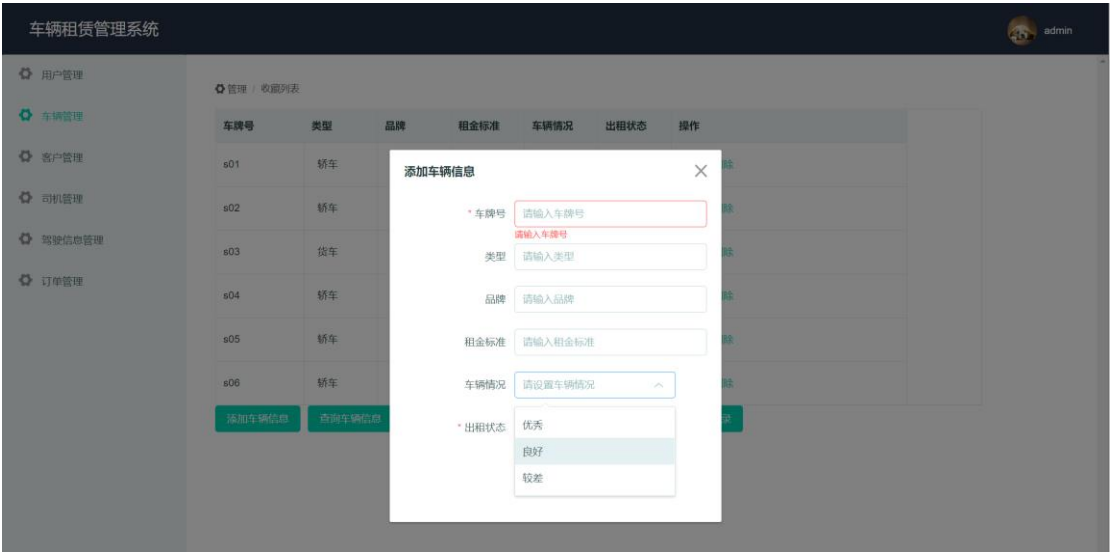
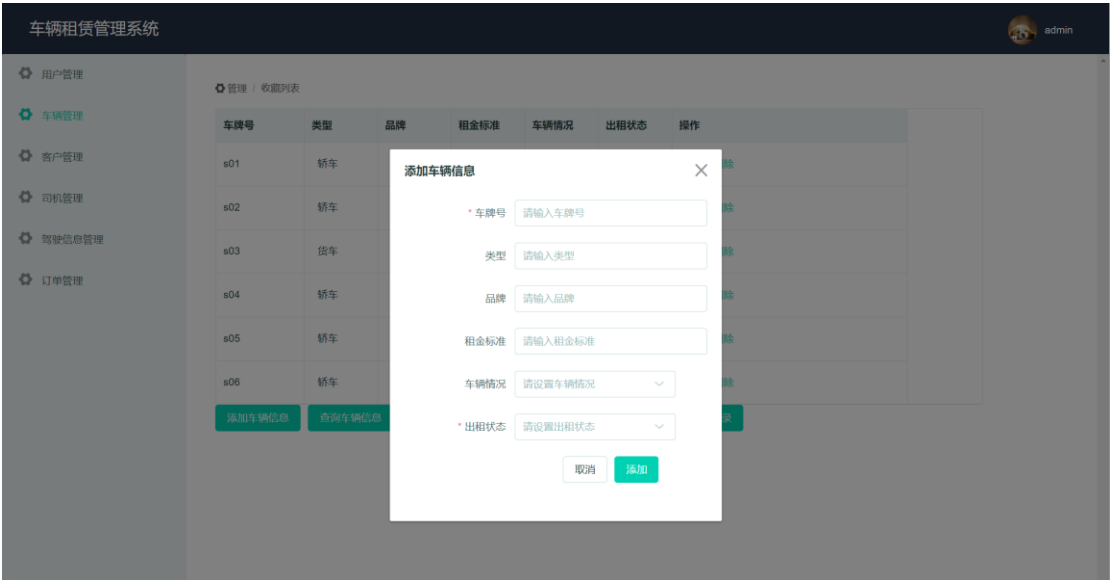


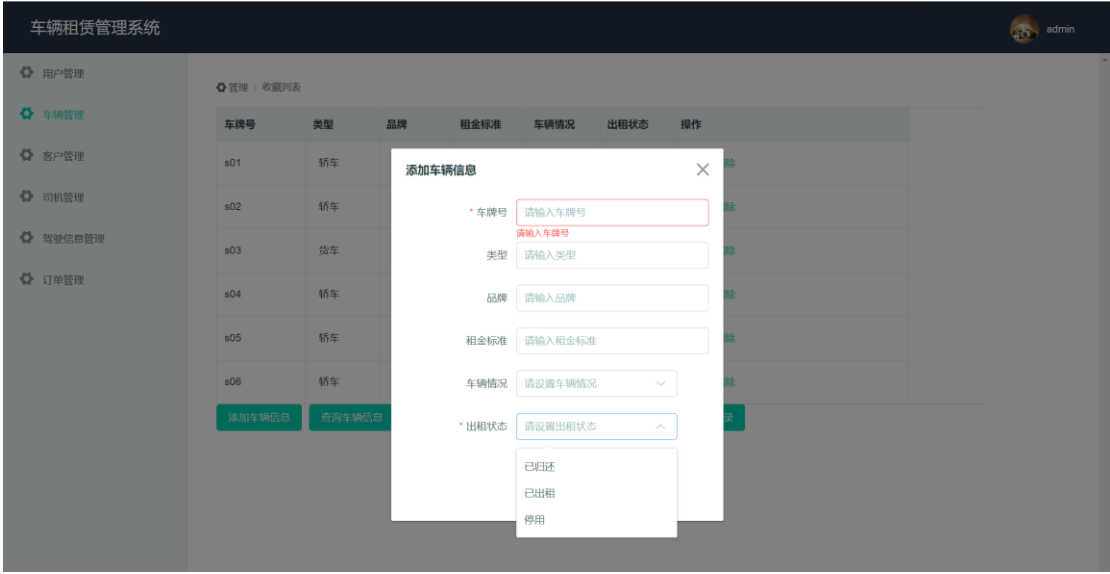


# 4.1.7 车辆信息管理模块界面



添加车辆信息:





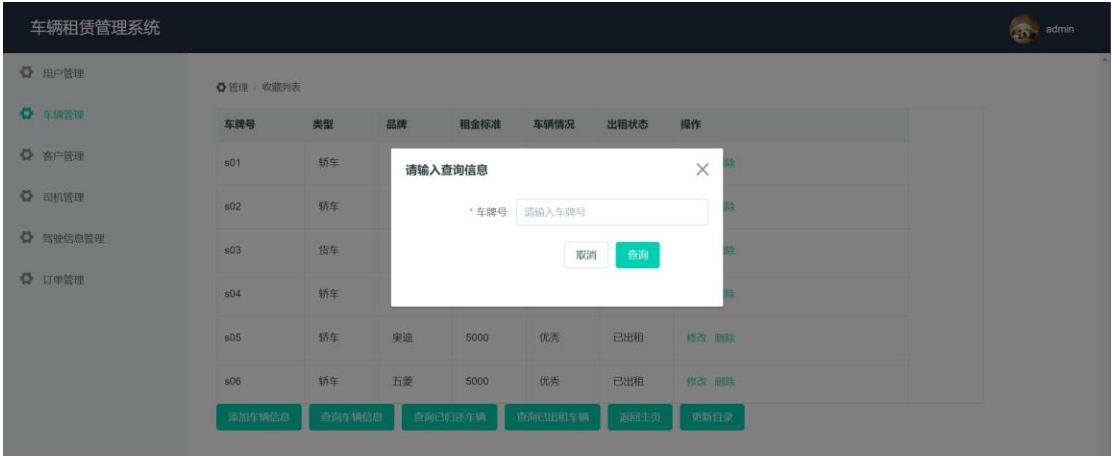
查询已归还车辆:



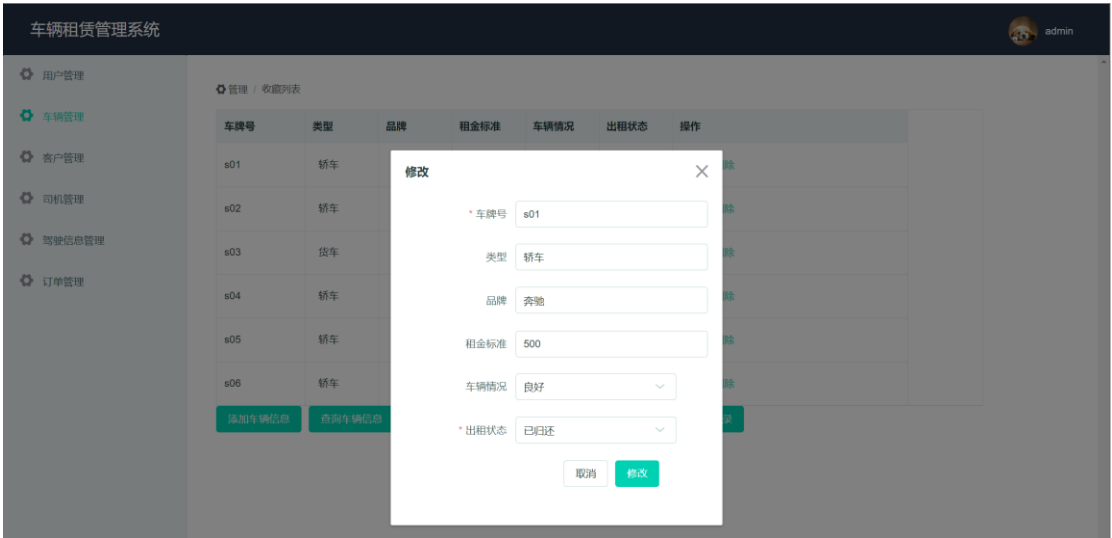
查询已出租车辆:



查询车辆信息:



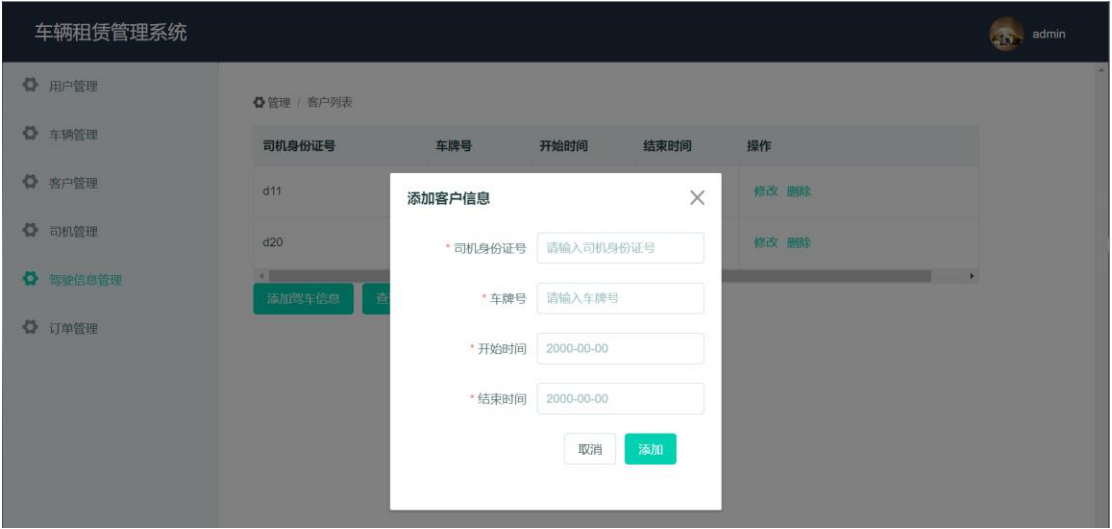
修改车辆信息:



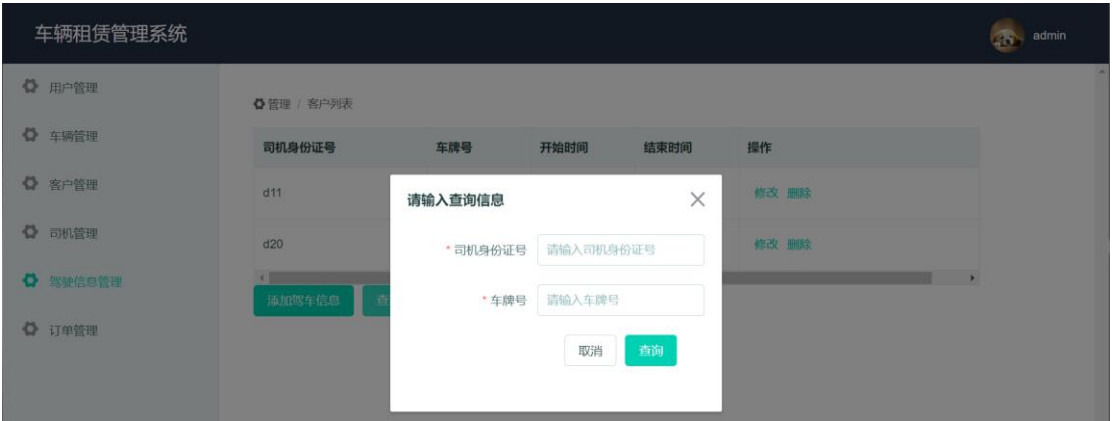
## 4.1.8 驾驶信息模块界面



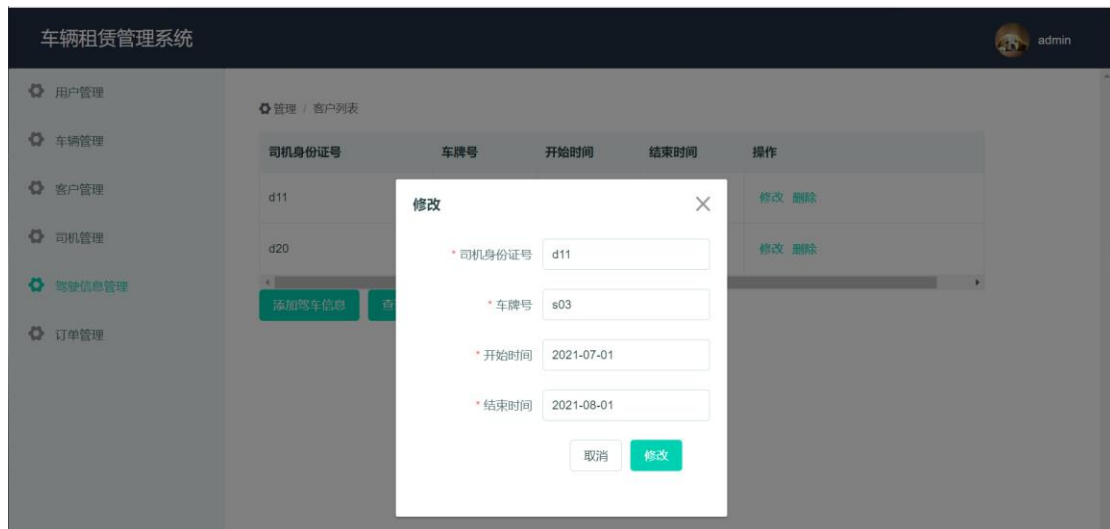
添加驾车信息:



查询驾车信息:



修改驾车信息:



## 4.1.9 订单信息管理模块界面

ID	客户身份证号	车牌号	司机身份证号	租赁模式	租赁日期	预计归还日期	实际归还日期	押金	实付款	违约金	订单状态	操作
1	c01	s02	未添加司机	私用	2020-06-01	2021-06-01	2021-05-01	50000	10000	0	提前归还	修改 删除 客户 司机 车辆
4	c11	s01	d05	公用	2020-01-01	2020-02-01	2020-03-01	50000	1000	500	逾期归还	修改 删除 客户 司机 车辆
5	c01	s01	d01	公用	2021-02-01	2021-03-01	2021-03-01	50000	6000	0	提前归还	修改 删除 客户 司机 车辆
11	c05	s02	未添加司机	公用	2020-06-01	2020-12-01		50000	0	0	未结算	修改 删除 客户 司机 车辆
15	c11	s03	未添加司机	公用	2021-06-01	2021-07-01		50000	0	0	未结算	修改 删除 客户 司机 车辆
16	c07	s05	d11	公用	2021-07-01	2021-08-01		10000	0	0	未结算	修改 删除 客户 司机 车辆
18	c02	s06	未添加司机	公用	2020-06-01	2020-07-01	2020-08-01	50000	5000	500	逾期归还	修改 删除 客户 司机 车辆
19	c20	s06	d20	公用	2022-06-01	2022-07-01		500	0	0	未结算	修改 删除 客户 司机 车辆

添加订单信息：

ID	客户身份证号	车牌号	司机身份证号	租赁模式	租赁日期	预计归还日期	实际归还日期	押金	实付款	违约金	订单状态	操作
1	c01	s02	未添加司机						10000	0	提前归还	修改 删除 客户 司机 车辆
4	c11	s01	d05						1000	500	逾期归还	修改 删除 客户 司机 车辆
5	c01	s01	d01						6000	0	提前归还	修改 删除 客户 司机 车辆
11	c05	s02	未添加司机						0	0	未结算	修改 删除 客户 司机 车辆
15	c11	s03	未添加司机						0	0	未结算	修改 删除 客户 司机 车辆
16	c07	s05	d11						0	0	未结算	修改 删除 客户 司机 车辆
18	c02	s06	未添加司机						5000	500	逾期归还	修改 删除 客户 司机 车辆
19	c20	s06	d20						0	0	未结算	修改 删除 客户 司机 车辆

### 添加订单信息

\* 客户身份证号  添加客户信息

\* 车牌号  添加车牌信息

司机身份证号  添加司机信息

租赁模式

租赁日期

预计归还日期

实际归还日期  以当前日期作为实际归还日期

订单状态

押金

实付款

违约金

取消 添加

添加客户信息：

车辆租赁管理系统

用户管理

车辆管理

客户管理

司机管理

租赁信息管理

订单管理

管理 / 订单列表

ID	客户身份证号	车牌号	司机身份证号	租赁模式	租赁日期	预计归还日期	实际归还日期	押金	实付款	违约金	订单状态	操作
1	c01	x02	未添加司机						10000	0	提前结算	修改 删除 客户 司机 车辆
4	c11	x01	x05						1000	500	逾期结算	修改 删除 客户 司机 车辆
5	x01	x01	x01						6000	0	按期结算	修改 删除 客户 司机 车辆
11	x05	x02	未添加司机						0	0	未结算	修改 删除 客户 司机 车辆
15	c11	x03	未添加司机						0	0	未结算	修改 删除 客户 司机 车辆
16	x07	x05	x11						0	0	未结算	修改 删除 客户 司机 车辆
18	x02	x06	未添加司机						5000	500	逾期结算	修改 删除 客户 司机 车辆
19	x20	x06	x20						0	0	未结算	修改 删除 客户 司机 车辆

添加订单信息

实时还款统计

押金统计

车辆订单信息

添加客户信息

\* ID

请输入身份证号

\* 姓名

请输入客户姓名

性别

请选择客户性别

年龄

0

\* 电话

请输入客户电话

地址

请输入客户地址

取消

添加

实际归还日期

请输入实际归还日期

订单状态

请选择订单状态

押金

0

实付款

0

违约金

0

取消

添加

添加司机信息：

车辆租赁管理系统

用户管理

车辆管理

客户管理

司机管理

租赁信息管理

订单管理

管理 / 订单列表

ID	客户身份证号	车牌号	司机身份证号	租赁模式	租赁日期	预计归还日期	实际归还日期	押金	实付款	违约金	订单状态	操作
1	c01	x02	未添加司机						10000	0	提前结算	修改 删除 客户 司机 车辆
4	c11	x01	x05						1000	500	逾期结算	修改 删除 客户 司机 车辆
5	x01	x01	x01						6000	0	按期结算	修改 删除 客户 司机 车辆
11	x05	x02	未添加司机						0	0	未结算	修改 删除 客户 司机 车辆
15	c11	x03	未添加司机						0	0	未结算	修改 删除 客户 司机 车辆
16	x07	x05	x11						0	0	未结算	修改 删除 客户 司机 车辆
18	x02	x06	未添加司机						5000	500	逾期结算	修改 删除 客户 司机 车辆
19	x20	x06	x20						0	0	未结算	修改 删除 客户 司机 车辆

添加订单信息

实时还款统计

押金统计

车辆订单信息

添加司机信息

\* ID

请输入身份证号

\* 姓名

请输入司机姓名

性别

请选择司机性别

年龄

0

\* 电话

请输入司机电话

执照

请输入执照编号

取消

添加

实际归还日期

请输入实际归还日期

订单状态

请选择订单状态

押金

0

实付款

0

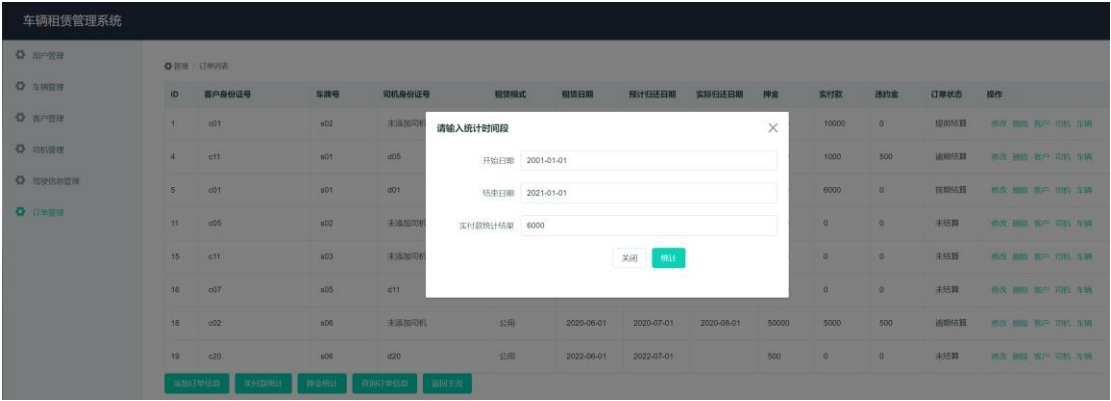
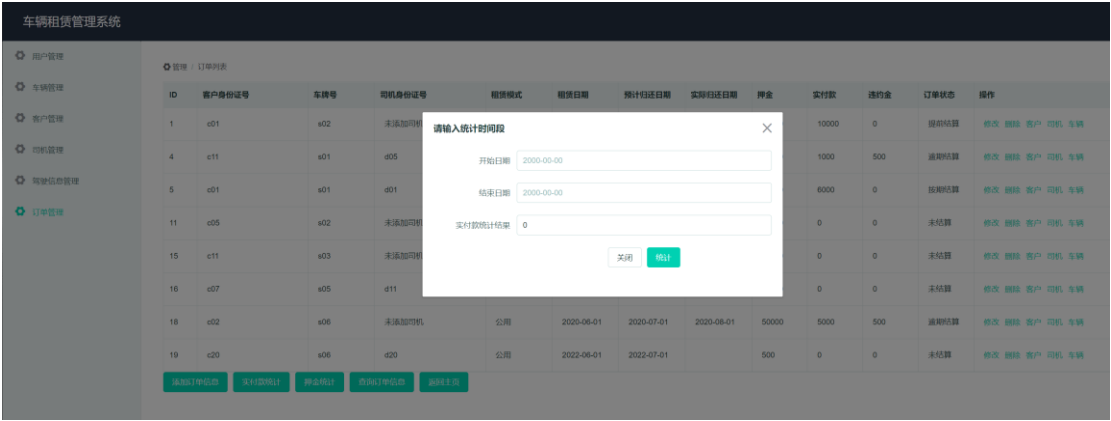
违约金

0

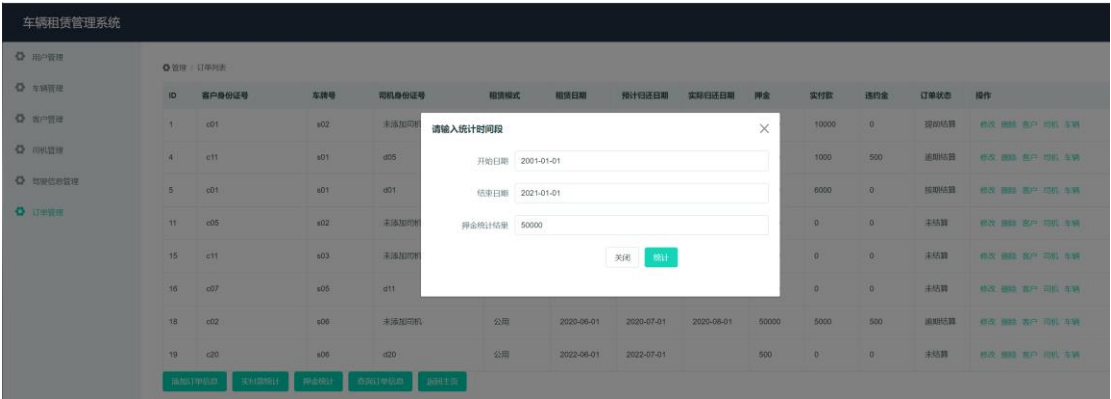
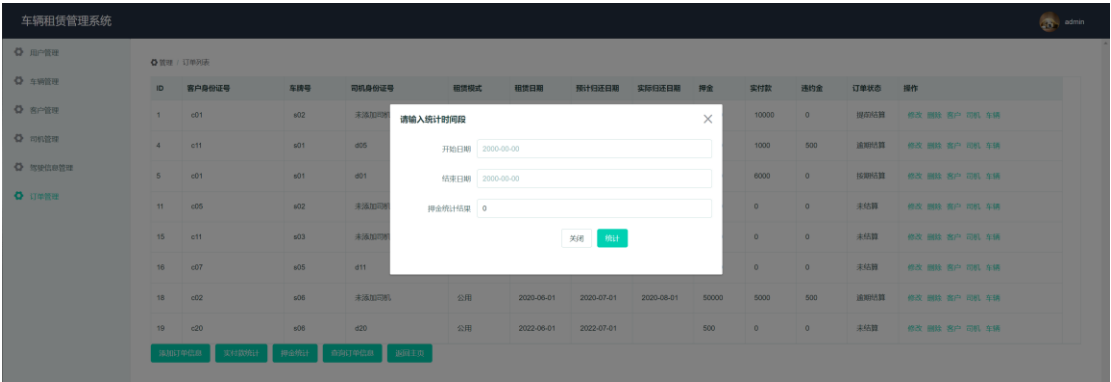
取消

添加

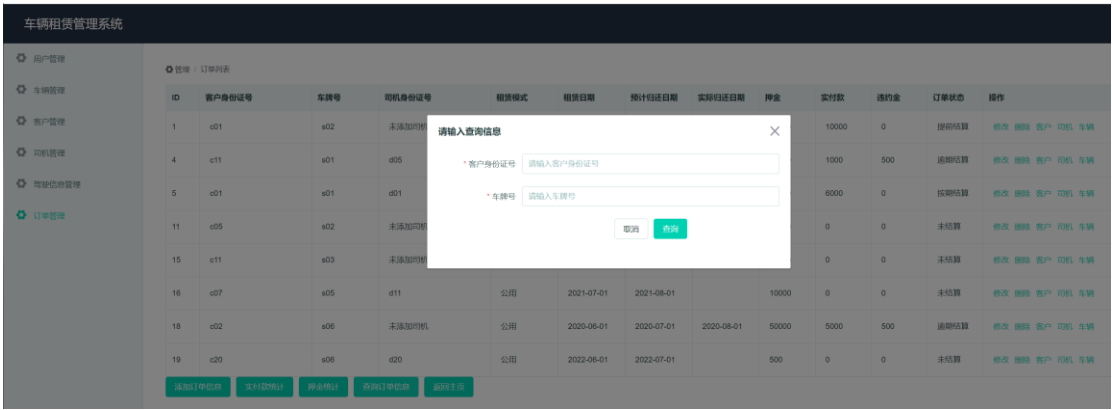
实付款统计：



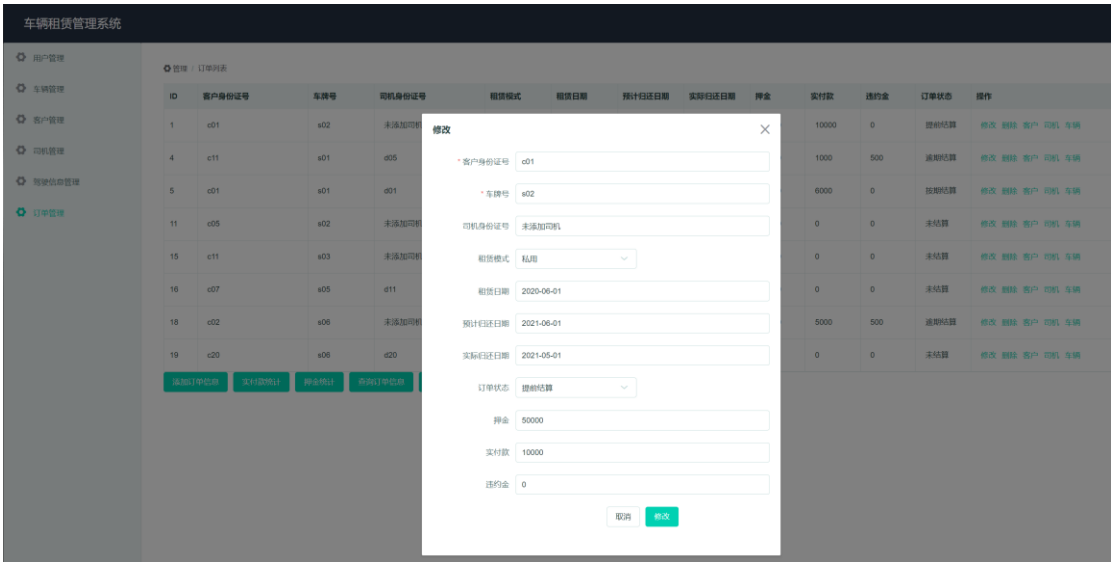
押金统计:



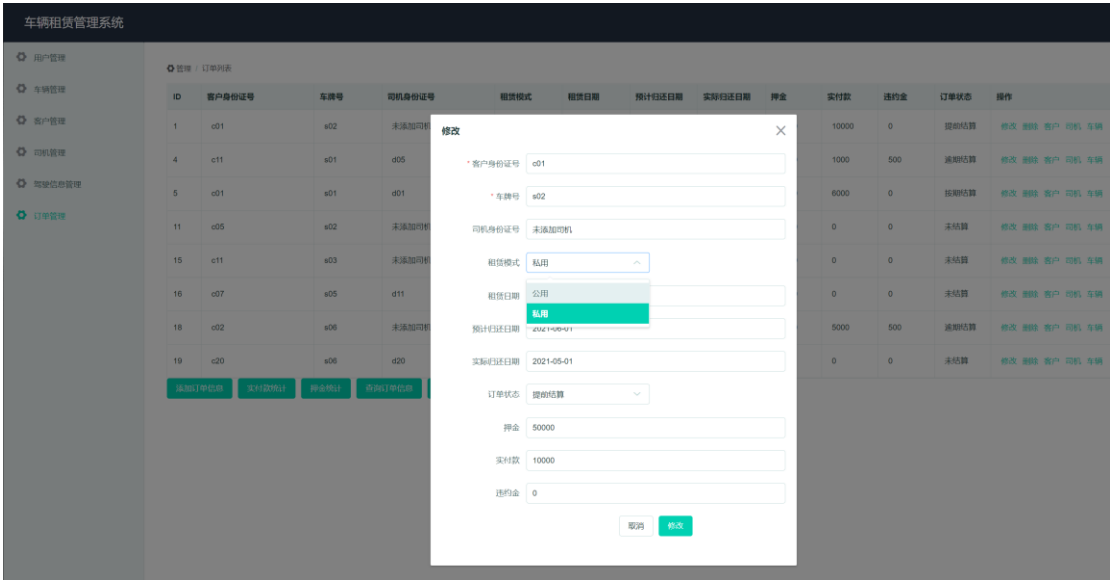
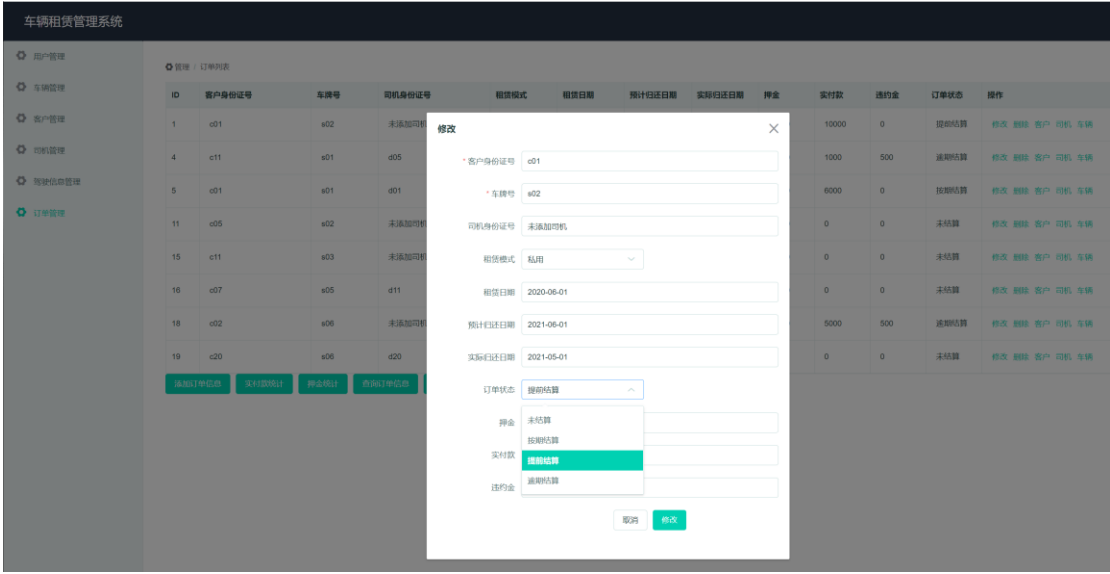
查询订单信息:



修改订单信息：







查看订单客户信息:



查看订单司机信息:

车辆租赁管理系统

用户管理

车辆管理

客户管理

司机管理

租赁信息管理

订单管理

管理 / 订单列表

ID	客户身份证号	车牌号	司机身份证号	租赁模式	租赁日期	预计归还日期	实际归还日期	押金	实付款	违约金	订单状态	操作
1	x01	x02	未添加司机						10000	0	提前结算	修改 删除 客户 司机 车辆
4	c11	x01	x05						1000	500	逾期结算	修改 删除 客户 司机 车辆
5	x01	x01	x01						6000	0	按期结算	修改 删除 客户 司机 车辆
11	x05	x02	未添加司机						0	0	未结算	修改 删除 客户 司机 车辆
15	c11	x03	未添加司机						0	0	未结算	修改 删除 客户 司机 车辆
16	x07	x05	x11						0	0	未结算	修改 删除 客户 司机 车辆
18	x02	x06	未添加司机						5000	500	逾期结算	修改 删除 客户 司机 车辆
19	x20	x06	x20						0	0	未结算	修改 删除 客户 司机 车辆

添加订单信息

实时数据统计

押金统计

逾期订单统计

司机信息查看

\* ID

未添加司机

\* 姓名

性别

年龄

0

\* 电话

照片

关闭

查看订单车辆信息：

车辆租赁管理系统

用户管理

车辆管理

客户管理

司机管理

租赁信息管理

订单管理

管理 / 订单列表

ID	客户身份证号	车牌号	司机身份证号	租赁模式	租赁日期	预计归还日期	实际归还日期	押金	实付款	违约金	订单状态	操作
1	x01	x02	未添加司机						10000	0	提前结算	修改 删除 客户 司机 车辆
4	c11	x01	x05						1000	500	逾期结算	修改 删除 客户 司机 车辆
5	x01	x01	x01						6000	0	按期结算	修改 删除 客户 司机 车辆
11	x05	x02	未添加司机						0	0	未结算	修改 删除 客户 司机 车辆
15	c11	x03	未添加司机						0	0	未结算	修改 删除 客户 司机 车辆
16	x07	x05	x11						0	0	未结算	修改 删除 客户 司机 车辆
18	x02	x06	未添加司机						5000	500	逾期结算	修改 删除 客户 司机 车辆
19	x20	x06	x20						0	0	未结算	修改 删除 客户 司机 车辆

添加订单信息

实时数据统计

押金统计

逾期订单统计

删除订单

车辆信息查看

\* 车牌号

x02

类型

轿车

品牌

奥迪

租金标准

5000

车辆情况

优秀

关闭

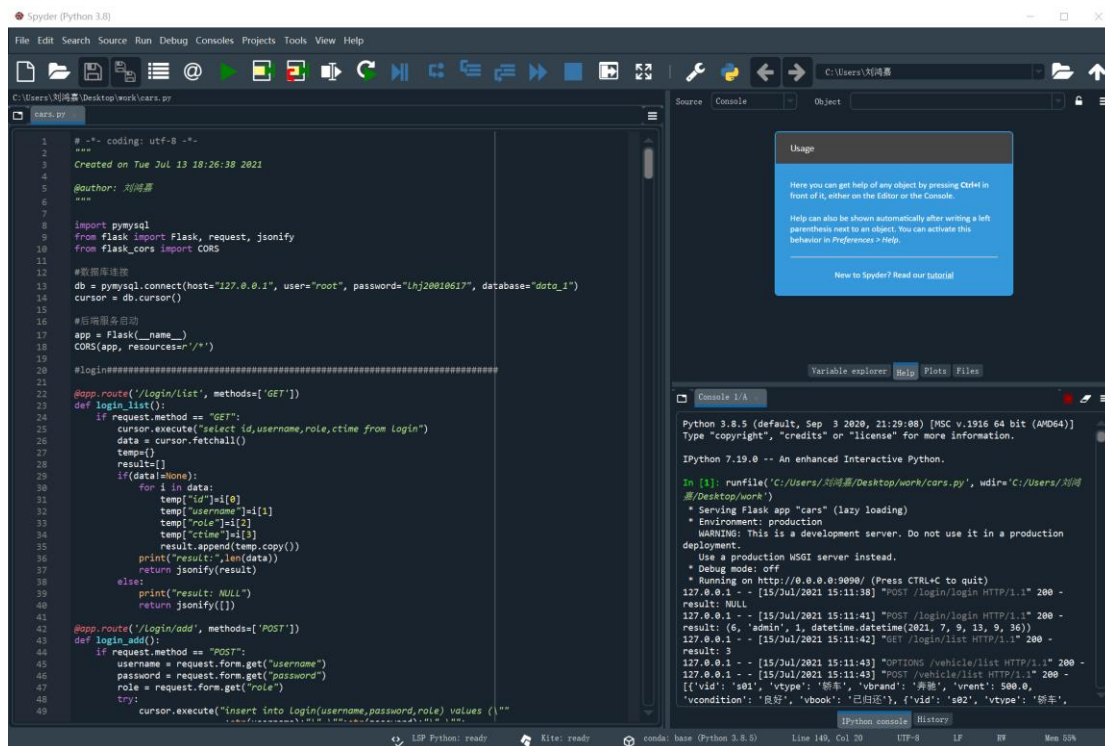
# 第五章 运行环境配置

## 5.1 后端环境

*Anaconda3 Spyder 4.1.5*

*Python 版本：3.8.5*

*Python 扩展包：pymysql, flask, flask\_cors*



## 5.2 Vue 搭建

```
C:\Users\刘鸿嘉>vue --version
@vue/cli 4.5.13
```

```
C:\Users\刘鸿嘉>vue ui
? Starting GUI...
? Ready on http://localhost:8000
```

## 5.3 MySQL 搭建

可视化界面工具: Navicat Premium 15

MySQL 版本: MySQL Server 8.0

# 第六章 后端操作

## 6.1 cars.py

```
# -*- coding: utf-8 -*-
"""
Created on Tue Jul 13 18:26:38 2021

@author: 刘鸿嘉
"""

import pymysql
from flask import Flask, request, jsonify
from flask_cors import CORS

#数据库连接
db = pymysql.connect(host="127.0.0.1", user="root", password="lhj20010617", database="data_1")
cursor = db.cursor()

#后端服务启动
app = Flask(__name__)
CORS(app, resources=r'/*')

#Login#####

#####

@app.route('/login/list', methods=['GET'])
def login_list():
    if request.method == "GET":
        cursor.execute("select id,username,role,ctime from login")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["id"]=i[0]
                temp["username"]=i[1]
                temp["role"]=i[2]
                temp["ctime"]=i[3]
                result.append(temp.copy())
```

```

        print("result:", len(data))
        return jsonify(result)
    else:
        print("result: NULL")
        return jsonify([])

@app.route('/login/add', methods=['POST'])
def login_add():
    if request.method == "POST":
        username = request.form.get("username")
        password = request.form.get("password")
        role = request.form.get("role")
        try:
            cursor.execute("insert into login(username,password,role) v
alues (\""+
                                +str(username)+"\", \""+str(password)+"\", \"
"+
                                str(role)+"\")")
            db.commit()
            print("add a new user successfully!")
            return "1"
        except Exception as e:
            print("add a new user failed:", e)
            db.rollback() # 发生错误就回滚
            return "-1"

@app.route('/login/login', methods=['POST'])
def login_login():
    if request.method == "POST":
        username = request.form.get("username")
        password = request.form.get("password")
        cursor.execute("select id,username,role,ctime from login where
username= \""+
                                +str(username)+"\" and password= \""+str(password
)+"\"")
        data = cursor.fetchone()
        if(data!=None):
            print("result:", data)
            jsongdata = {"id":str(data[0]), "username":str(data[1]),
                        "role":str(data[2]), "ctime":str(data[3])}
            return jsonify(jsongdata)
        else:
            print("result: NULL")
            jsongdata = {}

```

```

        return jsonify(jsondata)

@app.route('/login/update', methods=['POST'])
def login_update():
    if request.method == "POST":
        id = request.form.get("id")
        password = request.form.get("password")
        try:
            cursor.execute("update login set password=\'"+str(password)
                           +"\ " where id="+str(id))

            db.commit()
            print("update password successfully!")
            return "1"
        except Exception as e:
            print("update password failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

@app.route('/login/update_role', methods=['POST'])
def login_update_role():
    if request.method == "POST":
        id = request.form.get("id")
        role = request.form.get("role")
        try:
            cursor.execute("update login set role=\'"+str(role)
                           +"\ " where id="+str(id))

            db.commit()
            print("update role successfully!")
            return "1"
        except Exception as e:
            print("update role failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

@app.route('/login/del', methods=['POST'])
def login_del():
    if request.method == "POST":
        id = request.form.get("id")
        try:
            cursor.execute("delete from login where id="+str(id))
            db.commit()
            print("delete user successfully!")
            return "1"
        except Exception as e:

```

```

        print("delete user failed:",e)
        db.rollback() #发生错误就回滚
        return "-1"

#customer#####

#####

@app.route('/customer/list', methods=['POST'])
def customer_list():
    if request.method == "POST":
        cursor.execute("select cid,cname,csex,cage,cphone,caddress from
customer")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["cid"]=i[0]
                temp["cname"]=i[1]
                temp["csex"]=i[2]
                temp["cage"]=i[3]
                temp["cphone"]=i[4]
                temp["caddress"]=i[5]
                result.append(temp.copy())
            print("result:",data)
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

@app.route('/customer/add', methods=['POST'])
def customer_add():
    if request.method == "POST":
        cid = request.form.get("cid")
        cname = request.form.get("cname")
        csex = request.form.get("csex")
        cage = request.form.get("cage")
        cphone = request.form.get("cphone")
        caddress = request.form.get("caddress")
        try:
            cursor.execute("insert into customer(cid,cname,csex,cage,cp
hone,caddress) values (\"" +str(cid)
                                + "\",\"" +str(cname) + "\",\"" +str(csex) + "
\"",\"" +str(cage)

```

```

        + "\",\\"" + str(cphone) + "\",\\"" + str(address
    ) + "\"")
    db.commit()
    print("add a new customer successfully!")
    return "1"
except Exception as e:
    print("add a new customer failed:", e)
    db.rollback() # 发生错误就回滚
    return "-1"

@app.route('/customer/del', methods=['POST'])
def customer_del():
    if request.method == "POST":
        cid = request.form.get("cid")
        try:
            cursor.execute("delete from customer where cid=\\"" + str(cid)
+ "\"")

            db.commit()
            print("delete customer successfully!")
            return "1"
        except Exception as e:
            print("delete customer failed:", e)
            db.rollback() # 发生错误就回滚
            return "-1"

@app.route('/customer/update', methods=['POST'])
def customer_update():
    if request.method == "POST":
        cid = request.form.get("cid")
        cname = request.form.get("cname")
        csex = request.form.get("csex")
        cage = request.form.get("cage")
        cphone = request.form.get("cphone")
        caddress = request.form.get("caddress")
        try:
            cursor.execute("update customer set cid=\\"" + str(cid)
+ "\", cname=\\"" + str(cname) + "\", csex=\\""
+ str(csex) + "\", cage=\\"" + str(cage) + "\", cpho
ne=\\""
+ str(cphone) + "\", caddress=\\"" + str(caddress)
+ "\" where cid=\\"" + str(cid) + "\"")

            db.commit()
            print("update customer successfully!")
            return "1"

```



```

        except Exception as e:
            print("update customer failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

@app.route('/customer/search', methods=['POST'])
def customer_search():
    if request.method == "POST":
        cid = request.form.get("cid")
        cursor.execute("select cid,cname,csex,cage,cphone,caddress from
customer where cid=\ ""
                        + str(cid) + "\ """)
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["cid"]=i[0]
                temp["cname"]=i[1]
                temp["csex"]=i[2]
                temp["cage"]=i[3]
                temp["cphone"]=i[4]
                temp["caddress"]=i[5]
                result.append(temp.copy())
            print("result:",data)
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

#driver#####
#####

@app.route('/driver/list', methods=['POST'])
def driver_list():
    if request.method == "POST":
        cursor.execute("select did,dname,dsex,dage,dphone,ddlicense fro
m driver where did != \"未添加司机\"")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["did"]=i[0]

```

```

        temp["dname"]=i[1]
        temp["dsex"]=i[2]
        temp["dage"]=i[3]
        temp["dphone"]=i[4]
        temp["ddlicense"]=i[5]
        result.append(temp.copy())
    print("result:",len(data))
    return jsonify(result)
else:
    print("result: NULL")
    return jsonify([])

@app.route('/driver/add', methods=['POST'])
def driver_add():
    if request.method == "POST":
        did = request.form.get("did")
        dname = request.form.get("dname")
        dsex = request.form.get("dsex")
        dage = request.form.get("dage")
        dphone = request.form.get("dphone")
        ddlicense = request.form.get("ddlicense")
        try:
            cursor.execute( "insert into driver(did,dname,dsex,dage,dph
one,ddlicense) values (\""+
                        +str(did) + "\",\""+str(dname) + "\",\""+str
(dsex) + "\",\""+str(dage)
                        + "\",\""+str(dphone) + "\",\""+str(ddlicens
e) + "\")")
            db.commit()
            print("add driver successfully!")
            return "1"
        except Exception as e:
            print("add driver failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

@app.route('/driver/del', methods=['POST'])
def driver_del():
    if request.method == "POST":
        did = request.form.get("did")
        try:
            cursor.execute("delete from driver where did=\""+str(did)+"
\"")
            db.commit()

```

```

        print("delete driver successfully!")
        return "1"
    except Exception as e:
        print("delete driver failed:",e)
        db.rollback() #发生错误就回滚
        return "-1"

@app.route('/driver/update', methods=['POST'])
def driver_update():
    if request.method == "POST":
        did = request.form.get("did")
        dname = request.form.get("dname")
        dsex = request.form.get("dsex")
        dage = request.form.get("dage")
        dphone = request.form.get("dphone")
        ddlicense = request.form.get("ddlicense")
        try:
            cursor.execute("update driver set did='"+str(did)+"', dn
ame='"+
                                +str(dname)+"', dsex='"+str(dsex)+"', dag
e='"+
                                +str(dage)+"', dphone='"+str(dphone)
                                +"', ddlicense='"+str(ddlicense)+"' where
did='"+
                                +str(did)+"'")
            db.commit()
            print("update driver successfully!")
            return "1"
        except Exception as e:
            print("update driver failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

@app.route('/driver/search', methods=['POST'])
def driver_search():
    if request.method == "POST":
        did = request.form.get("did")
        cursor.execute("select did,dname,dsex,dage,dphone,ddlicense fro
m driver where did='"+
                                + str(did) + "'")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):

```

```

        for i in data:
            temp["did"]=i[0]
            temp["dname"]=i[1]
            temp["dsex"]=i[2]
            temp["dage"]=i[3]
            temp["dphone"]=i[4]
            temp["ddlicense"]=i[5]
            result.append(temp.copy())
        print("result:",data)
        return jsonify(result)
    else:
        print("result: NULL")
        return jsonify([])

#vehicle#####

#####

@app.route('/vehicle/list', methods=['POST'])
def vehicle_list():
    if request.method == "POST":
        cursor.execute("select vid,vtype,vbrand,vrent,vcondition,vbook
from vehicle where vdel = 0")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["vid"]=i[0]
                temp["vtype"]=i[1]
                temp["vbrand"]=i[2]
                temp["vrent"]=i[3]
                temp["vcondition"]=i[4]
                temp["vbook"]=i[5]
                result.append(temp.copy())
            print(result)
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

@app.route('/vehicle/list1', methods=['POST'])
def vehicle_list1():
    if request.method == "POST":

```

```

        cursor.execute("select vid,vtype,vbrand,vrent,vcondition,vbook
from vehicle where vdel = 0 and vbook = \"已归还\"")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["vid"]=i[0]
                temp["vtype"]=i[1]
                temp["vbrand"]=i[2]
                temp["vrent"]=i[3]
                temp["vcondition"]=i[4]
                temp["vbook"]=i[5]
                result.append(temp.copy())
            print(result)
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

```

```

@app.route('/vehicle/list2', methods=['POST'])

```

```

def vehicle_list2():

```

```

    if request.method == "POST":
        cursor.execute("select vid,vtype,vbrand,vrent,vcondition,vbook
from vehicle where vdel = 0 and vbook = \"已出租\"")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["vid"]=i[0]
                temp["vtype"]=i[1]
                temp["vbrand"]=i[2]
                temp["vrent"]=i[3]
                temp["vcondition"]=i[4]
                temp["vbook"]=i[5]
                result.append(temp.copy())
            print(result)
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

```

```

@app.route('/vehicle/list3', methods=['POST'])

```

```

def vehicle_list3():
    if request.method == "POST":
        vid = request.form.get("vid")
        cursor.execute("select vid,vtype,vbrand,vrent,vcondition,vbook
from vehicle where vid='"+str(vid) +"'")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["vid"]=i[0]
                temp["vtype"]=i[1]
                temp["vbrand"]=i[2]
                temp["vrent"]=i[3]
                temp["vcondition"]=i[4]
                temp["vbook"]=i[5]
                result.append(temp.copy())
            print("result:",len(data))
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

@app.route('/vehicle/add', methods=['POST'])
def vehicle_add():
    if request.method == "POST":
        vid = request.form.get("vid")
        vtype = request.form.get("vtype")
        vbrand = request.form.get("vbrand")
        vrent = request.form.get("vrent")
        vcondition = request.form.get("vcondition")
        vbook = request.form.get("vbook")
        try:
            cursor.execute("insert into vehicle(vid,vtype,vbrand,vrent,
vcondition,vbook) values (\''
                        +str(vid) + '\',\'"+str(vtype) + '\',\'"+str
(vbrand)
                        + '\',\'"+str(vrent) + '\',\'"+str(vconditio
n)
                        + '\',\'"+str(vbook) + "\'")")
            db.commit()
            print("add vehicle successfully!")
            return "1"
        except Exception as e:

```

```

        print("add vehicle failed:",e)
        db.rollback() #发生错误就回滚
        return "-1"

@app.route('/vehicle/del', methods=['POST'])
def vehicle_del():
    if request.method == "POST":
        vid = request.form.get("vid")
        try:
            cursor.execute("update vehicle set vdel = 1 where vid='"+s
tr(vid)+"'")
            db.commit()
            print("delete vehicle successfully!")
            return "1"
        except Exception as e:
            print("delete vehicle failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

@app.route('/vehicle/update', methods=['POST'])
def vehicle_update():
    if request.method == "POST":
        vid = request.form.get("vid")
        vtype = request.form.get("vtype")
        vbrand = request.form.get("vbrand")
        vrent = request.form.get("vrent")
        vcondition = request.form.get("vcondition")
        vbook = request.form.get("vbook")
        try:
            cursor.execute("update vehicle set vid='"+str(vid)
            +"'", vtype='"+str(vtype)+"'", vbrand='"+
            str(vbrand)+"'", vrent='"+str(vrent)
            +"'", vcondition='"+str(vcondition)+"'", vb
ook='"+
            str(vbook)+"' where vid='"+str(vid)+"'")
            db.commit()
            print("update vehicle successfully!")
            return "1"
        except Exception as e:
            print("update vehicle failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

@app.route('/vehicle/update_1', methods=['POST'])

```

```

def vehicle_update_1():
    if request.method == "POST":
        try:
            cursor.execute("update vehicle set vbook = \"已归还\" ")
            db.commit()
            cursor.execute("update vehicle set vbook = \"已出租\" where vid in (select vid from rent_order where state = \"未结算\")")
            db.commit()
            print("update_1 successfully!")
            return "1"
        except Exception as e:
            print("update_1 failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

#drive#####
#####

@app.route('/drive/list', methods=['POST'])
def drive_list():
    if request.method == "POST":
        cursor.execute("select vid,did,startdate,enddate from drive")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["vid"]=i[0]
                temp["did"]=i[1]
                temp["startdate"]=i[2]
                temp["enddate"]=i[3]
                result.append(temp.copy())
            print("result:",data)
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

@app.route('/drive/add', methods=['POST'])
def drive_add():
    if request.method == "POST":
        vid = request.form.get("vid")
        did = request.form.get("did")
        startdate = request.form.get("startdate")

```



```

        enddate = request.form.get("enddate")
        try:
            cursor.execute("insert into drive(vid,did,startdate,enddate
) values (\\""
                                + str(vid) + "\",\"" + str(did) + "\",\"" + str(s
tartdate)
                                + "\",\"" + str(enddate) + "\"")
            db.commit()
            print("add drive successfully!")
            return "1"
        except Exception as e:
            print("add drive failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

@app.route('/drive/del', methods=['POST'])
def drive_del():
    if request.method == "POST":
        vid = request.form.get("vid")
        did = request.form.get("did")
        try:
            cursor.execute("delete from drive where vid=\"" + str(vid)
                            + "\" and did=\"" + str(did) + "\"")
            db.commit()
            print("delete drive successfully!")
            return "1"
        except Exception as e:
            print("delete drive failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

@app.route('/drive/update', methods=['POST'])
def drive_update():
    if request.method == "POST":
        vid = request.form.get("vid")
        did = request.form.get("did")
        startdate = request.form.get("startdate")
        enddate = request.form.get("enddate")
        try:
            cursor.execute("update drive set vid=\"" + str(vid)
                            + "\", startdate=\"" + str(startdate) + "\", endd
ate=\""
                            + str(enddate) + "\" and did=\"" + str(did) + "\"")

```

```

        db.commit()
        print("update drive successfully!")
        return "1"
    except Exception as e:
        print("update drive failed:",e)
        db.rollback() #发生错误就回滚
        return "-1"

@app.route('/drive/search', methods=['POST'])
def drive_search():
    if request.method == "POST":
        vid = request.form.get("vid")
        did = request.form.get("did")
        cursor.execute("select vid,did,startdate,enddate from drive where vid=\\"+str(vid)
                                +\" \" and did=\\"+str(did) +\" \")
        data = cursor.fetchall()
        temp={}
        result=[]
        print(data)
        if(data!=None):
            for i in data:
                temp["vid"]=i[0]
                temp["did"]=i[1]
                temp["startdate"]=i[2]
                temp["enddate"]=i[3]
                result.append(temp.copy())
            print("result:",data)
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

#rent_order#####

#####

@app.route('/rent_order/list', methods=['POST'])
def rent_order_list():
    if request.method == "POST":
        cursor.execute("select * from rent_order")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):

```

```

        for i in data:
            temp["vid"]=i[0]
            temp["cid"]=i[1]
            temp["startdate"]=i[2]
            temp["enddate"]=i[3]
            temp["model"]=i[4]
            temp["adv_pay"]=i[5]
            temp["act_pay"]=i[6]
            temp["did"]=i[7]
            temp["rid"]=i[8]
            temp["bre_pay"]=i[9]
            temp["state"]=i[10]
            temp["act_enddate"]=i[11]
            result.append(temp.copy())
        print("result:",data)
        return jsonify(result)
    else:
        print("result: NULL")
        return jsonify([])

```

```

@app.route('/rent_order/add', methods=['POST'])

```

```

def rent_order_add():

```

```

    if request.method == "POST":
        vid = request.form.get("vid")
        cid = request.form.get("cid")
        did = request.form.get("did")
        startdate = request.form.get("startdate")
        enddate = request.form.get("enddate")
        act_enddate = request.form.get("act_enddate")
        model = request.form.get("model")
        adv_pay = request.form.get("adv_pay")
        act_pay = request.form.get("act_pay")
        bre_pay = request.form.get("bre_pay")
        state = request.form.get("state")
        try:
            cursor.execute("insert into rent_order(vid,cid,startdate,en
ddate,model,adv_pay,act_pay,did,bre_pay,state,act_enddate) values (\'"
                                + str(vid) + "\",\'"+str(cid) + "\",\'"
                                +str(startdate) + "\",\'"+str(enddate) + "\"
,\'"
                                + str(model) + "\",\'"+str(adv_pay) + "\",\'"
"
                                +str(act_pay) + "\",\'"+str(did)+ "\",\'"+st
r(bre_pay)

```

```

        + "\",\\"" + str(state) + "\",\\"" + str(act_enddat
e) + "\"")
        db.commit()
        print("add rent_order successfully!")
        return "1"
    except Exception as e:
        print("add rent_order failed:",e)
        db.rollback() #发生错误就回滚
        return "-1"

@app.route('/rent_order/del', methods=['POST'])
def rent_order_del():
    if request.method == "POST":
        vid = request.form.get("vid")
        cid = request.form.get("cid")
        rid = request.form.get("rid")
        try:
            cursor.execute("delete from rent_order where vid=\"\" + str(
vid)
                                + "\"" and cid=\"\" + str(cid) + "\"" and rid=\"
                                + str(rid)+\""")
            db.commit()
            print("delete successfully!")
            return "1"
        except Exception as e:
            print("delete failed:",e)
            db.rollback() #发生错误就回滚
            return "-1"

@app.route('/rent_order/update', methods=['POST'])
def rent_order_update():
    if request.method == "POST":
        rid = request.form.get("rid")
        vid = request.form.get("vid")
        cid = request.form.get("cid")
        did = request.form.get("did")
        startdate = request.form.get("startdate")
        enddate = request.form.get("enddate")
        act_enddate = request.form.get("act_enddate")
        model = request.form.get("model")
        adv_pay = request.form.get("adv_pay")
        act_pay = request.form.get("act_pay")
        bre_pay = request.form.get("bre_pay")

```

```

        state = request.form.get("state")
        sql = "update rent_order set vid=\'"+str(vid) +"\', cid=\'"+str(
(cid)+"\', did=\'"+str(did)
        sql = sql + "\', startdate=\'"+str(startdate)+"\', enddate=\'"+s
tr(enddate)+"\', model=\'"+str(model)
        sql = sql + "\', adv_pay=\'"+str(adv_pay)+"\', act_pay=\'"+str(a
ct_pay)+"\', bre_pay=\'"
        sql = sql + str(bre_pay)+"\', act_enddate=\'"+str(act_enddate)+
"\', state=\'"+str(state)+"\' where rid=\'"+str(rid)+"\'"
        # print(sql)
    try:
        cursor.execute(sql)
        db.commit()
        print("update rent_order successfully!")
        return "1"
    except Exception as e:
        print("update rent_order failed:",e)
        db.rollback() #发生错误就回滚
        return "-1"

@app.route('/rent_order_customer/list', methods=['POST'])
def rent_order_customer_list():
    if request.method == "POST":
        cid = request.form.get("cid")
        cursor.execute("select cid,cname,csex,cage,cphone,caddress from
customer where cid=\'"+str(cid)+"\'")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["cid"]=i[0]
                temp["cname"]=i[1]
                temp["csex"]=i[2]
                temp["cage"]=i[3]
                temp["cphone"]=i[4]
                temp["caddress"]=i[5]
                result.append(temp.copy())
            print("result:",len(data))
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

```

```

@app.route('/rent_order_driver/list', methods=['POST'])
def rent_order_driver_list():
    if request.method == "POST":
        did = request.form.get("did")
        sql = "select did,dname,dsex,dage,dphone,ddlicense from driver
where did=\""+str(did)+"\" "
        print(sql)
        cursor.execute(sql)
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["did"]=i[0]
                temp["dname"]=i[1]
                temp["dsex"]=i[2]
                temp["dage"]=i[3]
                temp["dphone"]=i[4]
                temp["ddlicense"]=i[5]
                result.append(temp.copy())
            print("result:",data)
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

@app.route('/rent_order_vehicle/list', methods=['POST'])
def rent_order_vehicle_list():
    if request.method == "POST":
        vid = request.form.get("vid")
        cursor.execute("select vid,vtype,vbrand,vrent,vcondition,vbook
from vehicle where vid=\""+str(vid)+"\" ")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["vid"]=i[0]
                temp["vtype"]=i[1]
                temp["vbrand"]=i[2]
                temp["vrent"]=i[3]
                temp["vcondition"]=i[4]
                temp["vbook"]=i[5]
                result.append(temp.copy())

```

```

        print("result:", len(data))
        return jsonify(result)
    else:
        print("result: NULL")
        return jsonify([])

@app.route('/rent_order/search', methods=['POST'])
def rent_order_search():
    if request.method == "POST":
        vid = request.form.get("vid")
        cid = request.form.get("cid")
        cursor.execute("select * from rent_order where vid=\'"+str(vid)
+"\' and cid=\'"+str(cid) +\'\'")
        data = cursor.fetchall()
        temp={}
        result=[]
        if(data!=None):
            for i in data:
                temp["vid"]=i[0]
                temp["cid"]=i[1]
                temp["startdate"]=i[2]
                temp["enddate"]=i[3]
                temp["model"]=i[4]
                temp["adv_pay"]=i[5]
                temp["act_pay"]=i[6]
                temp["did"]=i[7]
                temp["rid"]=i[8]
                temp["bre_pay"]=i[9]
                temp["state"]=i[10]
                temp["act_enddate"]=i[11]
                result.append(temp.copy())
            print("result:", len(data))
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

@app.route('/rent_order/adv_rent_sum', methods=['POST'])
def rent_order_adv_rent_sum():
    if request.method == "POST":
        _startdate = request.form.get("_startdate")
        _enddate = request.form.get("_enddate")
        sql = "select adv_pay from rent_order where startdate >= \'

```

```

        sql = sql + str(_startdate)+"\" and startdate <= \""+str(_endda
te)+"\" and state = \"未结算\""
        cursor.execute(sql)
        data = cursor.fetchall()
        print(sql)
        print(_startdate + "--" + _enddate)
        print(data)
        temp={}
        result=[]
        _sum = 0
        if(data!=None):
            for i in data:
                for j in i:
                    _sum += int(j)
                temp["adv_rent_sum"] = _sum
                result.append(temp.copy())
                print("result:",data)
            return jsonify(result)
        else:
            print("result: NULL")
            return jsonify([])

```

```

@app.route('/rent_order/act_rent_sum', methods=['POST'])
def rent_order_act_rent_sum():
    if request.method == "POST":
        _startdate = request.form.get("_startdate")
        _enddate = request.form.get("_enddate")
        sql = "select act_pay from rent_order where act_enddate >= \""
        sql = sql + str(_startdate) + "\" and act_enddate <= \""
        sql = sql + str(_enddate) + "\" and state != \"未结算\""
        cursor.execute(sql)
        data = cursor.fetchall()
        print(sql)
        print(_startdate + "--" + _enddate)
        print(data)
        temp={}
        result=[]
        _sum = 0
        if(data!=None):
            for i in data:
                for j in i:
                    _sum += int(j)
                temp["act_rent_sum"] = _sum
                result.append(temp.copy())

```

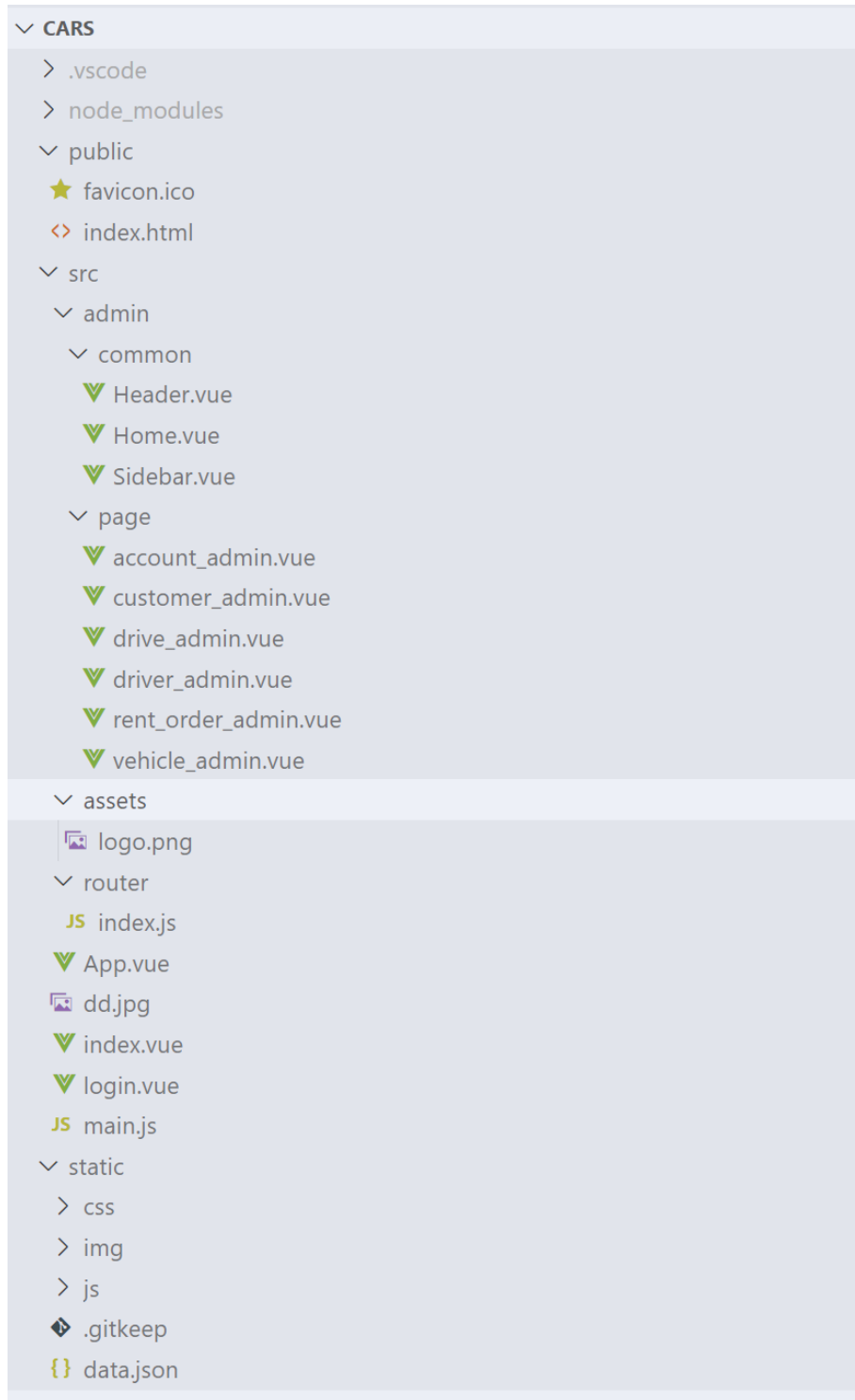


```
        print("result:", data)
        return jsonify(result)
    else:
        print("result: NULL")
        return jsonify([])

if __name__ == "__main__":
    app.run(host='0.0.0.0', port=9090)
    db.close()
    print("Good bye!")
```

# 第七章 前端操作

## 7.1 程序结构



## 7.2 main.js

```
import Vue from 'vue';
import App from './App';
import router from './router';
import axios from 'axios';
import ElementUI from 'element-ui';
//import 'element-ui/lib/theme-default/index.css'; // 默认主题
import 'element-ui/lib/theme-chalk/index.css';
import './static/css/theme-green/index.css'; // 浅绿色主题
import "babel-polyfill";
import VueResource from 'vue-resource'
import VueQuillEditor from 'vue-quill-editor'
import { Datetime } from 'vue-datetime'
// You need a specific Loader for CSS files
import 'vue-datetime/dist/vue-datetime.css'

Vue.use(Datetime)

Vue.use(VueQuillEditor);
Vue.use(VueResource);

Vue.use(ElementUI);
Vue.prototype.$axios = axios;
new Vue({
  el: '#app',
  router,
  render: h => h(App)
}).$mount('#app');

export default{ // 后端地址, 发布前要修改为后端服务器的 IP 地址
  url: "http://127.0.0.1:9090"
}
```

## 7.3 login.vue

```
<template>
  <div class="login-wrap">
    <div class="ms-title" style="font-size: 30px">我的收藏 - 用户登录
  <span style="font-size: 10px; margin-left: 73px;">
    <u></u></span>
  </div>
```

```

    <div class="demo-input-suffix ms-login">
      <el-
form :model="ruleForm" :rules="rules" ref="ruleForm" label-
width="70px" class="demo-ruleForm">
        <el-form-item prop="username" label="用户名" >
          <el-input v-
model="ruleForm.username" placeholder="请输入用户名"></el-input>
        </el-form-item>
        <el-form-item prop="password" label="密码" >
          <el-input v-
model="ruleForm.password" type="password" placeholder="请输入密码"></el-
input>
        </el-form-item>
        <el-form-item class="login-btn">
          <el-button type="primary" @click="submit()" >登
录</el-button>
          <el-button @click="cancel()" >返回</el-button>
        </el-form-item>
      </el-form>
    </div>
  </div>
</template>

<script>
import main from './main';
export default {
  data: function(){
    return {
      dialogVisible:false,
      ruleForm: {
        username: '',
        password: '',
      },
      rules: {
        username: [
          { required: true, message: '请输入用户名
', trigger: 'blur' }
        ],
        password: [
          { required: true, message: '请输入密码
', trigger: 'blur' }
        ],
      }
    }
  }
}

```

```

    },
    created(){
      localStorage.setItem('username','');
      localStorage.setItem('role','');
      localStorage.setItem('id','');
    },
    methods: {
      submit(){
        if(this.ruleForm.username===''){
          this.$message({type: 'info', message: '用户名必须输入! '});
        }else if(this.ruleForm.password===''){
          this.$message({type: 'info', message: '密码必须输入! '});
        }else{
          let crypto = require('crypto');
          const md5 = crypto.createHash('md5');
          md5.update(this.ruleForm.password);
          let md5password = md5.digest('hex');
          this.$http.post(main.url+"/login/login",
            { 'username': this.ruleForm.username, 'password': md5password },
            {
              headers: { 'Content-Type': 'application/x-www-form-urlencoded' },
              emulateJSON: true
            }).then(
              success => {
                if(success.data.id != null){
                  this.$message({type: 'success', message: '登录成功'});
                  localStorage.setItem('username',this.ruleForm.username);
                  localStorage.setItem('role',success.data.role);
                  localStorage.setItem('id',success.data.id);
                  this.$router.push({ path: '/admin' });
                  this.ruleForm.username='';
                }else{
                  this.$message({type: 'error', message: '用户名或密码错误'});
                }
                this.ruleForm.password='';
              }
            )
        }
      }
    }
  }

```

```

        this.dialogVisible=true;
    }
    );
}
},
cancel(){
    this.$router.push({ path: '/' });
}
}
}
</script>

<style scoped>
    .login-wrap{
        position: relative;
        background: url("/static/img/bg.jpg") no-repeat center;
        width:100%;
        height:100%;
    }
    .ms-title{
        position: absolute;
        top: 49%;
        width: 100%;
        margin-top: -230px;
        text-align: center;
        font-size: 14px;
        color: #fff;
        font-weight: bold;
    }
    .ms-login{
        position: absolute;
        left:48%;
        top:40%;
        width:300px;
        height:160px;
        margin:-150px 0 0 -190px;
        padding:40px;
        border-radius: 20px;
        /* background: #fff;*/
    }
    .login-btn{
        text-align: center;
    }
    .login-btn button{

```

```

        width:40%;
        height:35px;
        left:80px;
    }
</style>

```

## 7.4 Index.vue

```

<template>
  <div class="login-wrap">
    <div class="ms-title">
      <br>
      <br>
      车辆租赁管理系统
      <br>
      <br>
      <div>
        <el-
button type="primary" @click="submit()" style="font-size: 20px">请点击
此处进入</el-button>
      </div>
    </div>
  </div>
</template>

<script>
  export default {
    data: function(){
      return {
        dialogVisible:false,
        data:[]
      }
    },
    methods: {
      submit(){
        this.$router.push({ path: '/login' });
      }
    }
  }
</script>

<style scoped>
  .tab{

```

```

        position: absolute;
        top: 20%;
        left: 25%;
        width: 805px;
        height: 100%;
    }
    .login-wrap{
        position: relative;
        width:100%;
        height:100%;
        background: url("dd.jpg") no-repeat center center fixed;
        background-size: 100%;
    }
    .ms-title{
        position: absolute;
        top: 25%;
        left: 25%;
        width: 50%;
        height: 50%;
        border-radius:15px;
        text-align: center;
        font-size: 40px;
        color: #fff;
        font-weight: bold;
    }
    .login-btn button{
        position: absolute;
        width: 40%;
        height: 35px;
        right: 10%;
        top: 10%;
    }
</style>

```

## 7.5 App. vue

```

<template>
  <div id="app">
    <router-view></router-view>
  </div>
</template>

<style>

```



```

    @import "../static/css/main.css";
    @import "../static/css/color-dark.css";      /* 深色主题 */
    /*@import "../static/css/theme-green/color-green.css";  浅绿色主题
*/
</style>

```

## 7.6 Index.js

```

import Vue from 'vue'
import Router from 'vue-router'

Vue.use(Router);

export default new Router({
  routes: [
    {
      path: '/',
      component: resolve => require(['../index.vue'], resolve)
    },
    {
      path: '/login',
      component: resolve => require(['../login.vue'], resolve)
    },
    {
      path: '/admin',
      component: resolve => require(['../admin/common/Home.vue'],
      resolve),
      children: [
        {
          path: '/account_admin',
          component: resolve => require(['../admin/page/account_admin.vue'], resolve)
        },
        {
          path: '/vehicle_admin',
          component: resolve => require(['../admin/page/vehicle_admin.vue'], resolve)
        },
        {
          path: '/customer_admin',
          component: resolve => require(['../admin/page/customer_admin.vue'], resolve)
        }
      ]
    }
  ]
})

```

```

        {
          path: '/driver_admin',
          component: resolve => require(['../admin/page/drive
r_admin.vue'], resolve)
        },
        {
          path: '/drive_admin',
          component: resolve => require(['../admin/page/drive
_admin.vue'], resolve)
        },
        {
          path: '/rent_order_admin',
          component: resolve => require(['../admin/page/rent_
order_admin.vue'], resolve)
        },
      ]
    }
  ]
})

```

## 7.7 vehicle\_admin.vue

```

<template>
  <div>
    <div class="crumbs">
      <el-breadcrumb separator="/">
        <el-breadcrumb-item>
          <i class="el-icon-setting"></i> 管理</el-breadcrumb-item>
        >
        <el-breadcrumb-item>收藏列表</el-breadcrumb-item>
      </el-breadcrumb>
    </div>
    <div>
      <el-
table :data="data" border style="width: 90%" ref="multipleTable">
        <el-table-column>
          label="车牌号"
          prop="vid"
          width="120px"
        ></el-table-column>
        <el-table-column>
          label="类型"
          prop="vtype"

```

```

        width="100px"
      </el-table-column>
      <el-table-column
        label="品牌"
        prop="vbrand"
        width="100px"
      </el-table-column>
      <el-table-column
        label="租金标准"
        prop="vrent"
        width="100px"
      </el-table-column>
      <el-table-column
        label="车辆情况"
        prop="vcondition"
        width="100px"
      </el-table-column>
      <el-table-column
        label="出租状态"
        prop="vbook"
        width="100px"
      </el-table-column>
      <el-table-column label="操作" width="320px">
        <template slot-scope="scope" width="100px">
          <el-button type="text" @click="openedit(scope.row)">修改
        </el-button>
          <el-button type="text" @click="del_fav(scope.row)">删除</el-
button>
        </template>
      </el-table-column> </el-table>
    <el-button type="primary" @click="addfavorite()" align="center"
    >添加车辆信息</el-button>
    <el-button type="primary" @click="search()" align="center"
    >查询车辆信息</el-button>
    <el-button type="primary" @click="search_1()" align="center"
    >查询已归还车辆</el-button>
    <el-button type="primary" @click="search_2()" align="center"
    >查询已出租车辆</el-button>
    <el-button type="primary" @click="init()" align="center"
    >返回主页</el-button>
    <el-button type="primary" @click="update_v()" align="center"
    >更新目录</el-button>
  >
</div>

```

```

    <el-dialog width="30%" title="修改
" :visible.sync="dialogFormVisible">
    <div>
      <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
        <el-form-item label="车牌号" prop="vid">
          <el-input v-model="form.vid" placeholder="请输入车牌号"></el-
input>
        </el-form-item>
        <el-form-item label="类型" prop="vtype">
          <el-input v-model="form.vtype" placeholder="请输入类型
"></el-input>
        </el-form-item>
        <el-form-item label="品牌" prop="vbrand">
          <el-input v-model="form.vbrand" placeholder="请输入品牌
"></el-input>
        </el-form-item>
        <el-form-item label="租金标准" prop="vrent">
          <el-input
            v-model="form.vrent"
            placeholder="请输入租金标准"
          ></el-input>
        </el-form-item>
        <el-form-item label="车辆情况" prop="vcondition">
          <el-select v-model="form.vcondition" placeholder="请设置车辆
情况">
            <el-option
              v-for="item in options1"
              :key="item.value"
              :label="item.label"
              :value="item.value"
            >
          </el-option>
        </el-select>
      </el-form-item>
      <el-form-item label="出租状态" prop="vbook">
        <el-select v-model="form.vbook" placeholder="请设置出租状态">
          <el-option
            v-for="item in options"
            :key="item.value"
            :label="item.label"
            :value="item.value"
          >
        </el-option>

```

```

        </el-select>
      </el-form-item>
      <el-form-item style="text-align: center">
        <el-button @click="dialogFormVisibleled = false">取消</el-
button>
        <el-button type="primary" @click="submit('form')">修改</el-
button>
      </el-form-item>
    </el-form>
  </div>
</el-dialog>
<el-dialog
  width="30%"
  title="添加车辆信息"
  :visible.sync="dialogFormVisibleled1"
>
  <div>
    <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
      <el-form-item label="车牌号" prop="vid">
        <el-input v-model="form.vid" placeholder="请输入车牌号"></el-
input>
      </el-form-item>
      <el-form-item label="类型" prop="vtype">
        <el-input v-model="form.vtype" placeholder="请输入类型
"></el-input>
      </el-form-item>
      <el-form-item label="品牌" prop="vbrand">
        <el-input v-model="form.vbrand" placeholder="请输入品牌
"></el-input>
      </el-form-item>
      <el-form-item label="租金标准" prop="vrent">
        <el-input
          v-model="form.vrent"
          placeholder="请输入租金标准"
        ></el-input>
      </el-form-item>
      <el-form-item label="车辆情况" prop="vcondition">
        <el-select v-model="form.vcondition" placeholder="请设置车辆
情况">
          <el-option
            v-for="item in options1"
            :key="item.value"
            :label="item.label"

```

```

        :value="item.value"
      >
    </el-option>
  </el-select>
</el-form-item>
<el-form-item label="出租状态" prop="vbook">
  <el-select v-model="form.vbook" placeholder="请设置出租状态">
    <el-option
      v-for="item in options"
      :key="item.value"
      :label="item.label"
      :value="item.value"
    >
  </el-option>
</el-select>
</el-form-item>
<el-form-item style="text-align: center">
  <el-button @click="dialogFormVisible1 = false">取消</el-
button>
  <el-button type="primary" @click="addnew(form)">添加</el-
button>
</el-form-item>
</el-form>
</div>
</el-dialog>
<el-dialog
  width="30%"
  title="请输入查询信息"
  :visible.sync="dialogFormVisible5"
>
  <div>
    <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
      <el-form-item label="车牌号" prop="vid">
        <el-input v-model="form.vid" placeholder="请输入车牌号"></el-
input>
      </el-form-item>
      <el-form-item style="text-align: center">
        <el-button @click="dialogFormVisible5 = false">取消</el-
button>
        <el-button type="primary" @click="searchorder(form)"
        >查询</el-button
      >
    </el-form-item>

```

```

        </el-form>
      </div>
    </el-dialog>
  </div>
</template>

<script>
import main from "../../main";
export default {
  data: function () {
    return {
      data: [],
      options: [
        { value: "已归还", label: "已归还" },
        { value: "已出租", label: "已出租" },
        { value: "停用", label: "停用" },
      ],
      options1: [
        { value: "优秀", label: "优秀" },
        { value: "良好", label: "良好" },
        { value: "较差", label: "较差" },
      ],
      dialogFormVisible: false,
      dialogFormVisible1: false,
      dialogFormVisible5: false,
      form: {
        vid: "",
        vtype: "",
        vbrand: "",
        vrent: "",
        vcondition: "",
        vbook: "",
      },
      rules: {
        vid: [{ required: true, message: "请输入车牌号", trigger: "blur" }],
        vbook: [{ required: true, message: "请输入出租状态", trigger: "blur" }],
      },
    };
  },
  created() {
    if (localStorage.getItem("username") === "") {
      this.$router.replace("/login");
    }
  }
};

```

```

    } else {
        this.init();
    }
},
methods: {
    init_form() {
        this.form = {
            vid: "",
            vtype: "",
            vbrand: "",
            vrent: "",
            vcondition: "",
            vbook: "",
        };
    },
    init() {
        this.$http
            .post(main.url + "/vehicle/list", {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            })
            .then((success) => {
                this.data = success.data;
                this.init_form();
            });
    },
    update_v() {
        this.$http
            .post(main.url + "/vehicle/update_1", {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            })
            .then((success) => {
                console.log(success);
                this.init();
            });
    },
    search_1() {
        this.$http
            .post(main.url + "/vehicle/list1", {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },

```



```

        emulateJSON: true,
    })
    .then((success) => {
        this.data = success.data;
    });
},
search_2() {
    this.$http
        .post(main.url + "/vehicle/list2", {
            headers: { "Content-Type": "application/x-www-form-
urlencoded" },
            emulateJSON: true,
        })
        .then((success) => {
            this.data = success.data;
        });
},
search() {
    this.dialogFormVisible5 = true;
    this.init_form();
},
searchorder() {
    this.$http
        .post(
            main.url + "/vehicle/list3",
            {
                vid: this.form.vid,
            },
            {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            }
        )
        .then((success) => {
            this.data = success.data;
            this.dialogFormVisible5 = false;
        });
},
openedit(row) {
    // 修改收藏内容, 按钮按下后
    this.dialogFormVisible = true;
    this.form = {
        vid: row.vid,
    },

```

```

        vtype: row.vtype,
        vbrand: row.vbrand,
        vrent: row.vrent,
        vcondition: row.vcondition,
        vbook: row.vbook,
    };
},
submit() {
    // 修改收藏内容, 提交
    this.$http
        .post(
            main.url + "/vehicle/update",
            {
                vid: this.form.vid,
                vtype: this.form.vtype,
                vbrand: this.form.vbrand,
                vrent: this.form.vrent,
                vcondition: this.form.vcondition,
                vbook: this.form.vbook,
            },
            {
                headers: { "Content-Type": "application/x-www-form-urlencoded" },
                emulateJSON: true,
            }
        )
        .then((success) => {
            console.log(success);
            this.$message({ type: "success", message: "修改成功" });
            this.dialogFormVisible = false;
            this.init();
        });
},
addfavorite() {
    this.dialogFormVisible1 = true;
    this.init_form();
},
addnew() {
    // 添加新收藏
    if (this.form.vid === "")
        this.$message({ type: "error", message: "车牌号! " });
    else if (this.form.vbook === "")
        this.$message({ type: "error", message: "出租状态! " });
    else {

```

```

        this.$http
        .post(
            main.url + "/vehicle/add",
            {
                vid: this.form.vid,
                vtype: this.form.vtype,
                vbrand: this.form.vbrand,
                vrent: this.form.vrent,
                vcondition: this.form.vcondition,
                vbook: this.form.vbook,
            },
            {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            }
        )
        .then((success) => {
            console.log(success);
            this.$message({ type: "success", message: "添加成功" });
            this.init();
        });
        this.dialogFormVisible1 = false;
    },
    del_fav(row) {
        // 删除收藏记录
        this.$confirm("请确认是否要删除该车辆信息!", "提示", {
            confirmButtonText: "确定",
            cancelButtonText: "取消",
            type: "warning",
        })
        .then(() => {
            this.$http
            .post(
                main.url + "/vehicle/del",
                { vid: row.vid },
                {
                    headers: {
                        "Content-Type": "application/x-www-form-urlencoded",
                    },
                    emulateJSON: true,
                }
            )
        })
    }
}

```

```

        .then((success) => {
            console.log(success);
            this.$message({ type: "success", message: "已删除" });
            this.init();
        });
    })
    .catch(() => {
        this.$message({ type: "info", message: "已取消" });
    });
},
},
};
</script>

```

## 7.8 rent\_order\_admin.vue

```

<template>
  <div>
    <div class="crumbs">
      <el-breadcrumb separator="/">
        <el-breadcrumb-item>
          <i class="el-icon-setting"></i> 管理</el-breadcrumb-item>
        <el-breadcrumb-item>订单列表</el-breadcrumb-item>
      </el-breadcrumb>
    </div>
    <div>
      <el-
table :data="data" border style="width: 100%" ref="multipleTable">
      <el-table-column label="ID" prop="rid" width="60px"></el-table-
column>
      <el-table-column
        label="客户身份证号"
        prop="cid"
        width="200px"
      ></el-table-column>
      <el-table-column
        label="车牌号"
        prop="vid"
        width="120px"
      ></el-table-column>
      <el-table-column
        label="司机身份证号"

```

```
        prop="did"
        width="200px"
    ></el-table-column>
    <el-table-column
        label="租赁模式"
        prop="model"
        width="120px"
    ></el-table-column>
    <el-table-column
        label="租赁日期"
        prop="startdate"
        width="120px"
    ></el-table-column>
    <el-table-column
        label="预计归还日期"
        prop="enddate"
        width="120px"
    ></el-table-column>
    <el-table-column
        label="实际归还日期"
        prop="act_enddate"
        width="120px"
    ></el-table-column>
    <el-table-column
        label="押金"
        prop="adv_pay"
        width="100px"
    ></el-table-column>
    <el-table-column
        label="实付款"
        prop="act_pay"
        width="100px"
    ></el-table-column>
    <el-table-column
        label="违约金"
        prop="bre_pay"
        width="100px"
    ></el-table-column>
    <el-table-column
        label="订单状态"
        prop="state"
        width="100px"
    ></el-table-column>
    <el-table-column label="操作" width="320px">
```

```

        <template slot-scope="scope" width="100px">
            <el-button type="text" @click="openedit(scope.row)">修改
        </el-button>
            <el-button type="text" @click="del_driver(scope.row)"
                >删除</el-button>
            >
            <el-button type="text" @click="display_customer(scope.row)"
                >客户</el-button>
            >
            <el-button type="text" @click="display_driver(scope.row)"
                >司机</el-button>
            >
            <el-button type="text" @click="display_vehicle(scope.row)"
                >车辆</el-button>
            >
        </template>
    </el-table-column> </el-table>
    ><el-button type="primary" @click="addorder()" align="center"
        >添加订单信息</el-button>
    ><el-button type="primary" @click="act_money()" align="center"
        >实付款统计</el-button>
    ><el-button type="primary" @click="adv_money()" align="center"
        >押金统计</el-button>
    ><el-button type="primary" @click="search()" align="center"
        >查询订单信息</el-button>
    ><el-button type="primary" @click="init()" align="center"
        >返回主页</el-button>
    >
    <template> </template>
</div>
<!------->
<el-dialog width="30%" title="修改
" :visible.sync="dialogFormVisible">
    <div>
        <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
            <el-form-item label="客户身份证号" prop="cid">
                <el-input
                    v-model="form.cid"
                    placeholder="请输入客户身份证号"
                ></el-input>
            </el-form-item>
            <el-form-item label="车牌号" prop="vid">

```

```
input>      <el-input v-model="form.vid" placeholder="请输入车牌号"></el-
input>

</el-form-item>
<el-form-item label="司机身份证号" prop="did">
  <el-input
    v-model="form.did"
    placeholder="请输入司机身份证号"
  ></el-input>
</el-form-item>
<el-form-item label="租赁模式" prop="model">
  <el-select v-model="form.model" placeholder="请设置租赁模式">
    <el-option
      v-for="item in options"
      :key="item.value"
      :label="item.label"
      :value="item.value"
    >
  </el-option>
</el-select>
</el-form-item>
<el-form-item label="租赁日期" prop="startdate">
  <el-input
    v-model="form.startdate"
    placeholder="请输入租赁日期"
  ></el-input>
</el-form-item>
<el-form-item label="预计归还日期" prop="enddate">
  <el-input
    v-model="form.enddate"
    placeholder="请输入预计归还日期"
  ></el-input>
</el-form-item>
<el-form-item label="实际归还日期" prop="act_enddate">
  <el-input
    v-model="form.act_enddate"
    placeholder="请输入实际归还日期"
  ></el-input>
</el-form-item>
<el-form-item label="订单状态" prop="state">
  <el-select v-model="form.state" placeholder="请设置订单状态">
    <el-option
      v-for="item in options2"
      :key="item.value"
      :label="item.label"
```

```

        :value="item.value"
      >
    </el-option>
  </el-select>
</el-form-item>
<el-form-item label="押金" prop="adv_pay">
  <el-input
    v-model="form.adv_pay"
    placeholder="请输入押金"
  ></el-input>
</el-form-item>
<el-form-item label="实付款" prop="act_pay">
  <el-input
    v-model="form.act_pay"
    placeholder="请输入实付款"
  ></el-input>
</el-form-item>
<el-form-item label="违约金" prop="bre_pay">
  <el-input
    v-model="form.bre_pay"
    placeholder="请输入违约金"
  ></el-input>
</el-form-item>
<el-form-item style="text-align: center">
  <el-button @click="dialogFormVisible = false">取消</el-
button>
  <el-button type="primary" @click="submit('form')">修改</el-
button>
</el-form-item>
</el-form>
</div>
</el-dialog>
<!------->
<!------->
<el-dialog
  width="30%"
  title="添加订单信息"
  :visible.sync="dialogFormVisible1"
>
  <div>
    <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
      <el-form-item label="客户身份证号" prop="cid">
        <el-input

```



```

        v-model="form.cid"
        placeholder="请输入客户身份证号"
    ></el-input>
</el-form-item>
<el-form-item style="text-align: center">
    <el-button type="primary" @click="add_customer(form)"
        >添加客户信息</el-button
    >
</el-form-item>
<el-form-item label="车牌号" prop="vid">
    <el-input v-model="form.vid" placeholder="请输入车牌号"></el-
input>
</el-form-item>
<el-form-item label="司机身份证号" prop="did">
    <el-input
        v-model="form.did"
        placeholder="请输入司机身份证号"
    ></el-input>
</el-form-item>
<el-form-item style="text-align: center">
    <el-button type="primary" @click="add_driver(form)"
        >添加司机信息</el-button
    >
</el-form-item>
<el-form-item label="租赁模式" prop="model">
    <el-select v-model="form.model" placeholder="请设置租赁模式">
        <el-option
            v-for="item in options"
            :key="item.value"
            :label="item.label"
            :value="item.value"
        >
        </el-option>
    </el-select>
</el-form-item>
<el-form-item label="租赁日期" prop="startdate">
    <el-input
        v-model="form.startdate"
        placeholder="请输入租赁日期"
    ></el-input>
</el-form-item>
<el-form-item label="预计归还日期" prop="enddate">
    <el-input
        v-model="form.enddate"

```

```
        placeholder="请输入预计归还日期"
    ></el-input>
</el-form-item>
<el-form-item label="实际归还日期" prop="act_enddate">
    <el-input
        v-model="form.act_enddate"
        placeholder="请输入实际归还日期"
    ></el-input>
</el-form-item>
<el-form-item style="text-align: center">
    <el-button type="primary" @click="add_drive(form)"
        >以当前租赁归还日期添加驾车信息</el-button
    >
</el-form-item>
<el-form-item label="订单状态" prop="state">
    <el-select v-model="form.state" placeholder="请设置订单状态">
        <el-option
            v-for="item in options2"
            :key="item.value"
            :label="item.label"
            :value="item.value"
        >
        </el-option>
    </el-select>
</el-form-item>
<el-form-item label="押金" prop="adv_pay">
    <el-input
        v-model="form.adv_pay"
        placeholder="请输入押金"
    ></el-input>
</el-form-item>
<el-form-item label="实付款" prop="act_pay">
    <el-input
        v-model="form.act_pay"
        placeholder="请输入实付款"
    ></el-input>
</el-form-item>
<el-form-item label="违约金" prop="bre_pay">
    <el-input
        v-model="form.bre_pay"
        placeholder="请输入违约金"
    ></el-input>
</el-form-item>
<el-form-item style="text-align: center">
```

```

        <el-button @click="dialogFormVisibleled1 = false">取消</el-
button>
        <el-button type="primary" @click="addnew(form)">添加</el-
button>
    </el-form-item>
</el-form>
</div>
</el-dialog>
<!------->
<!------->
<el-dialog
width="30%"
title="客户信息查看"
:visible.sync="dialogFormVisibleled2"
>
    <div>
        <el-form
            :model="form"
            :rules="rules_cus"
            ref="form"
            label-width="100px"
        >
            <el-form-item label="ID" prop="cid">
                <el-input v-model="form.cid" placeholder=""></el-input>
            </el-form-item>
            <el-form-item label="姓名" prop="cname">
                <el-input v-model="form.cname" placeholder=""></el-input>
            </el-form-item>
            <el-form-item label="性别" prop="csex">
                <el-input v-model="form.csex" placeholder=""></el-input>
            </el-form-item>
            <el-form-item label="年龄" prop="cage">
                <el-input v-model="form.cage" placeholder=""></el-input>
            </el-form-item>
            <el-form-item label="电话" prop="cphone">
                <el-input v-model="form.cphone" placeholder=""></el-input>
            </el-form-item>
            <el-form-item label="地址" prop="caddress">
                <el-input v-model="form.caddress" placeholder=""></el-
input>
            </el-form-item>
            <el-form-item style="text-align: center">
                <el-
button type="primary" @click="dialogFormVisibleled2 = false"

```

```

        >关闭</el-button
      >
    </el-form-item>
  </el-form>
</div>
</el-dialog>
<!------->
<!------->
<el-dialog
  width="30%"
  title="司机信息查看"
  :visible.sync="dialogFormVisibleled3"
>
  <div>
    <el-form
      :model="form"
      :rules="rules_driver"
      ref="form"
      label-width="80px"
    >
      <el-form-item label="ID" prop="did">
        <el-input v-model="form.did" placeholder=""></el-input>
      </el-form-item>
      <el-form-item label="姓名" prop="dname">
        <el-input v-model="form.dname" placeholder=""></el-input>
      </el-form-item>
      <el-form-item label="性别" prop="dsex">
        <el-input v-model="form.dsex" placeholder=""></el-input>
      </el-form-item>
      <el-form-item label="年龄" prop="dage">
        <el-input v-model="form.dage" placeholder=""></el-input>
      </el-form-item>
      <el-form-item label="电话" prop="dphone">
        <el-input v-model="form.dphone" placeholder=""></el-input>
      </el-form-item>
      <el-form-item label="执照" prop="ddlicense">
        <el-input v-model="form.ddlicense" placeholder=""></el-
input>
      </el-form-item>
      <el-form-item style="text-align: center">
        <el-
button type="primary" @click="dialogFormVisibleled3 = false"
        >关闭</el-button
      >
    </div>
  </el-dialog>

```

```

        </el-form-item>
      </el-form>
    </div>
  </el-dialog>
  <!------->
  <el-dialog
    width="30%"
    title="车辆信息查看"
    :visible.sync="dialogFormVisible4"
  >
    <div>
      <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
        <el-form-item label="车牌号" prop="vid">
          <el-input v-model="form.vid" placeholder=""></el-input>
        </el-form-item>
        <el-form-item label="类型" prop="vtype">
          <el-input v-model="form.vtype" placeholder=""></el-input>
        </el-form-item>
        <el-form-item label="品牌" prop="vbrand">
          <el-input v-model="form.vbrand" placeholder=""></el-input>
        </el-form-item>
        <el-form-item label="租金标准" prop="vrent">
          <el-input v-model="form.vrent" placeholder=""></el-input>
        </el-form-item>
        <el-form-item label="车辆情况" prop="vcondition">
          <el-input v-model="form.vcondition" placeholder=""></el-
input>
        </el-form-item>
        <el-form-item style="text-align: center">
          <el-
button type="primary" @click="dialogFormVisible4 = false"
            >关闭</el-button>
        >
      </el-form-item>
    </el-form>
  </div>
</el-dialog>
<!------->
  <el-dialog
    width="30%"
    title="请输入查询信息"

```

```

        :visible.sync="dialogFormVisible5"
      >
        <div>
          <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
            <el-form-item label="客户身份证号" prop="cid">
              <el-input
                v-model="form.cid"
                placeholder="请输入客户身份证号"
              ></el-input>
            </el-form-item>
            <el-form-item label="车牌号" prop="vid">
              <el-input v-model="form.vid" placeholder="请输入车牌号"></el-
input>
            </el-form-item>
            <el-form-item style="text-align: center">
              <el-button @click="dialogFormVisible5 = false">取消</el-
button>
              <el-button type="primary" @click="searchorder(form)"
                >查询</el-button>
            </el-form-item>
          </el-form>
        </div>
      </el-dialog>
    <!------->
    <el-dialog
      width="30%"
      title="请输入统计时间段"
      :visible.sync="dialogFormVisible6"
    >
      <div>
        <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
          <el-form-item label="开始日期" prop="_startdate">
            <el-input
              v-model="form._startdate"
              placeholder="2000-00-00"
            ></el-input>
          </el-form-item>
          <el-form-item label="结束日期" prop="_enddate">
            <el-input
              v-model="form._enddate"

```

```

        placeholder="2000-00-00"
      ></el-input>
    </el-form-item>
    <el-form-item label="押金统计结果" prop="adv_rent_sum">
      <el-input
        v-model="form.adv_rent_sum"
        placeholder="请按下“统计”按钮"
      ></el-input>
    </el-form-item>
    <el-form-item style="text-align: center">
      <el-button @click="dialogFormVisible6 = false">关闭</el-
button>
      <el-button type="primary" @click="adv_money_sum(form)"
        >统计</el-button
    >
    </el-form-item>
  </el-form>
</div>
</el-dialog>
<!------->
<el-dialog
  width="30%"
  title="添加司机信息"
  :visible.sync="dialogFormVisible7"
>
  <div>
    <el-form
      :model="form"
      :rules="rules_driver"
      ref="form"
      label-width="80px"
    >
      <el-form-item label="ID" prop="did">
        <el-input
          v-model="form.did"
          placeholder="请输入身份证号"
        ></el-input>
      </el-form-item>
      <el-form-item label="姓名" prop="dname">
        <el-input
          v-model="form.dname"
          placeholder="请输入司机姓名"
        ></el-input>
    </el-form>
  </div>

```

```

</el-form-item>
<el-form-item label="性别" prop="dsex">
  <el-select v-model="form.dsex" placeholder="请设置司机性别">
    <el-option
      v-for="item in options1"
      :key="item.value"
      :label="item.label"
      :value="item.value"
    >
  </el-option>
</el-select>
</el-form-item>
<el-form-item label="年龄" prop="dage">
  <el-input
    v-model="form.dage"
    placeholder="请输入司机年龄"
  ></el-input>
</el-form-item>
<el-form-item label="电话" prop="dphone">
  <el-input
    v-model="form.dphone"
    placeholder="请输入司机电话"
  ></el-input>
</el-form-item>
<el-form-item label="执照" prop="ddlicense">
  <el-input
    v-model="form.ddlicense"
    placeholder="请输入执照编号"
  ></el-input>
</el-form-item>
<el-form-item style="text-align: center">
  <el-button @click="dialogFormVisible7 = false">取消</el-
button>
  <el-button type="primary" @click="add_driver_new(form)"
    >添加</el-button>
  >
</el-form-item>
</el-form>
</div>
</el-dialog>
<!--
----->
<el-dialog
  width="30%"

```



```
title="添加客户信息"
:visible.sync="dialogFormVisibleled8"
>
<div>
  <el-form :model="form" :rules="rules_cus" ref="form" label-
width="80px">
    <el-form-item label="ID" prop="cid">
      <el-input
        v-model="form.cid"
        placeholder="请输入身份证号"
      ></el-input>
    </el-form-item>
    <el-form-item label="姓名" prop="cname">
      <el-input
        v-model="form.cname"
        placeholder="请输入客户姓名"
      ></el-input>
    </el-form-item>
    <el-form-item label="性别" prop="csex">
      <el-select v-model="form.csex" placeholder="请设置客户性别">
        <el-option
          v-for="item in options1"
          :key="item.value"
          :label="item.label"
          :value="item.value"
        >
      </el-option>
    </el-select>
    </el-form-item>
    <el-form-item label="年龄" prop="cage">
      <el-input
        v-model="form.cage"
        placeholder="请输入客户年龄"
      ></el-input>
    </el-form-item>
    <el-form-item label="电话" prop="cphone">
      <el-input
        v-model="form.cphone"
        placeholder="请输入客户电话"
      ></el-input>
    </el-form-item>
    <el-form-item label="地址" prop="caddress">
      <el-input
        v-model="form.caddress"
```

```

        placeholder="请输入客户地址"
      ></el-input>
    </el-form-item>
    <el-form-item style="text-align: center">
      <el-button @click="dialogFormVisible8 = false">取消</el-
button>
      <el-button type="primary" @click="add_customer_new(form)"
      >添加</el-button>
    >
  </el-form-item>
</el-form>
</div>
</el-dialog>
<!------->
<el-dialog
  width="30%"
  title="请输入统计时间段"
  :visible.sync="dialogFormVisible9"
  >
  <div>
    <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
      <el-form-item label="开始日期" prop="_startdate">
        <el-input
          v-model="form._startdate"
          placeholder="2000-00-00"
        ></el-input>
      </el-form-item>
      <el-form-item label="结束日期" prop="_enddate">
        <el-input
          v-model="form._enddate"
          placeholder="2000-00-00"
        ></el-input>
      </el-form-item>
      <el-form-item label="实付款统计结果" prop="act_rent_sum">
        <el-input
          v-model="form.act_rent_sum"
          placeholder="请按下“统计”按钮"
        ></el-input>
      </el-form-item>
      <el-form-item style="text-align: center">
        <el-button @click="dialogFormVisible9 = false">关闭</el-
button>

```

```

        <el-button type="primary" @click="act_money_sum(form)"
        >统计</el-button
    >
    </el-form-item>
</el-form>
</div>
</el-dialog>
</div>
</template>

<script>
import main from ".././main";
export default {
  data: function () {
    return {
      data: [],
      options: [
        { value: "公用", label: "公用" },
        { value: "私用", label: "私用" },
      ],
      options1: [
        { value: "男", label: "男" },
        { value: "女", label: "女" },
      ],
      options2: [
        { value: "未结算", label: "未结算" },
        { value: "按期结算", label: "按期结算" },
        { value: "提前结算", label: "提前结算" },
        { value: "逾期结算", label: "逾期结算" },
      ],
      dialogFormVisibleled: false,
      dialogFormVisibleled1: false,
      dialogFormVisibleled2: false,
      dialogFormVisibleled3: false,
      dialogFormVisibleled4: false,
      dialogFormVisibleled5: false,
      dialogFormVisibleled6: false,
      dialogFormVisibleled7: false,
      dialogFormVisibleled8: false,
      dialogFormVisibleled9: false,
      form: {
        cid: "",
        vid: "",
        did: "",

```

```
model: "",
startdate: "",
enddate: "",
act_enddate: "",
adv_pay: 0,
act_pay: 0,
_enddate: "",
_startdate: "",
cname: "",
csex: "",
cage: 0,
cphone: "",
caddress: "",
dname: "",
dsex: "",
dage: 0,
dphone: "",
ddlicense: "",
vtype: "",
vbrand: "",
vrent: "",
vcondition: "",
adv_rent_sum: 0,
act_rent_sum: 0,
rid: 0,
bre_pay: 0,
state: "",
vbook: "",
},
rules: {
  cid: [
    { required: true, message: "请输入客户身份证号", trigger: "blur" },
  ],
  vid: [{ required: true, message: "请输入车牌号", trigger: "blur" }],
},
rules_driver: {
  did: [{ required: true, message: "请输入身份证号", trigger: "blur" }],
  dname: [{ required: true, message: "请输入司机姓名", trigger: "blur" }],
  dphone: [
    { required: true, message: "请输入司机电话", trigger: "blur" },
  ],
}
```

```

    ],
  },
  rules_cus: {
    cid: [{ required: true, message: "请输入身份证号", trigger: "blur" }],
    cname: [{ required: true, message: "请输入客户姓名", trigger: "blur" }],
    cphone: [
      { required: true, message: "请输入客户电话", trigger: "blur" },
    ],
  },
};
},
created() {
  if (localStorage.getItem("username") === "") {
    this.$router.replace("/login");
  } else {
    this.init();
  }
},
methods: {
  init_form() {
    this.form = {
      cid: "",
      vid: "",
      did: "",
      model: "",
      startdate: "",
      enddate: "",
      act_enddate: "",
      adv_pay: 0,
      act_pay: 0,
      _enddate: "",
      _startdate: "",
      cname: "",
      csex: "",
      cage: 0,
      cphone: "",
      caddress: "",
      dname: "",
      dsex: "",
      dage: 0,
      dphone: "",
      ddlicense: "",
    };
  }
}

```

```

        vtype: "",
        vbrand: "",
        vrent: "",
        vcondition: "",
        adv_rent_sum: 0,
        act_rent_sum: 0,
        rid: 0,
        bre_pay: 0,
        state: "",
        vbook: "",
    };
},
init() {
    this.$http
        .post(main.url + "/rent_order/list", {
            headers: { "Content-Type": "application/x-www-form-
urlencoded" },
            emulateJSON: true,
        })
        .then((success) => {
            this.data = success.data;
            this.init_form();
        });
},
adv_money() {
    this.dialogFormVisible6 = true;
    this.init_form();
},
adv_money_sum() {
    this.$http
        .post(
            main.url + "/rent_order/adv_rent_sum",
            {
                _startdate: this.form._startdate,
                _enddate: this.form._enddate,
            },
            {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            }
        )
        .then((success) => {
            this.form.adv_rent_sum = success.data[0].adv_rent_sum;

```

```

    });
},
act_money() {
    this.dialogFormVisible9 = true;
    this.init_form();
},
act_money_sum() {
    this.$http
        .post(
            main.url + "/rent_order/act_rent_sum",
            {
                _startdate: this.form._startdate,
                _enddate: this.form._enddate,
            },
            {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            }
        )
        .then((success) => {
            this.form.act_rent_sum = success.data[0].act_rent_sum;
        });
},
display_customer(row) {
    //显示客户信息
    this.dialogFormVisible2 = true;
    this.$http
        .post(
            main.url + "/rent_order_customer/list",
            {
                cid: row.cid,
            },
            {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            }
        )
        .then((success) => {
            this.form = success.data[0];
        });
},
display_driver(row) {

```

```

// 显示司机信息
this.dialogFormVisible3 = true;
this.$http
  .post(
    main.url + "/rent_order_driver/list",
    {
      did: row.did,
    },
    {
      headers: { "Content-Type": "application/x-www-form-
urlencoded" },
      emulateJSON: true,
    }
  )
  .then((success) => {
    this.form = success.data[0];
  });
},
display_vehicle(row) {
  // 显示司机信息
  this.dialogFormVisible4 = true;
  this.$http
    .post(
      main.url + "/rent_order_vehicle/list",
      {
        vid: row.vid,
      },
      {
        headers: { "Content-Type": "application/x-www-form-
urlencoded" },
        emulateJSON: true,
      }
    )
    .then((success) => {
      this.form = success.data[0];
    });
},
openedit(row) {
  // 修改内容, 按钮按下后
  this.dialogFormVisible = true;
  this.form = {
    rid: row.rid,
    cid: row.cid,
    vid: row.vid,
  }
}

```



```

        did: row.did,
        model: row.model,
        startdate: row.startdate,
        enddate: row.enddate,
        act_enddate: row.act_enddate,
        adv_pay: row.adv_pay,
        act_pay: row.act_pay,
        bre_pay: row.bre_pay,
        state: row.state,
    };
},
submit() {
    // 修改收藏内容, 提交
    if (this.form.did === "") this.form.did = "未添加司机";
    this.$http
        .post(
            main.url + "/rent_order/update",
            {
                rid: this.form.rid,
                cid: this.form.cid,
                vid: this.form.vid,
                did: this.form.did,
                model: this.form.model,
                startdate: this.form.startdate,
                enddate: this.form.enddate,
                act_enddate: this.form.act_enddate,
                adv_pay: this.form.adv_pay,
                act_pay: this.form.act_pay,
                bre_pay: this.form.bre_pay,
                state: this.form.state,
            },
            {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            }
        )
        .then((success) => {
            console.log(success);
            this.$message({ type: "success", message: "修改成功" });
            this.dialogFormVisible = false;
            this.init();
        });
},

```

```

search() {
  this.dialogFormVisible5 = true;
  this.init_form();
},
searchorder() {
  this.$http
    .post(
      main.url + "/rent_order/search",
      {
        cid: this.form.cid,
        vid: this.form.vid,
      },
      {
        headers: { "Content-Type": "application/x-www-form-urlencoded" },
        emulateJSON: true,
      }
    )
    .then((success) => {
      this.data = success.data;
      this.dialogFormVisible5 = false;
    });
},
add_drive() {
  if (this.form.did === "")
    this.$message({ type: "error", message: "司机身份证号为空！" });
  else if (this.form.vid === "")
    this.$message({ type: "error", message: "车牌号为空！" });
  else if (this.form.startdate === "")
    this.$message({ type: "error", message: "租赁日期为空！" });
  else if (this.form.enddate === "")
    this.$message({ type: "error", message: "归还日期为空！" });
  else {
    this.$http
      .post(
        main.url + "/drive/add",
        {
          did: this.form.did,
          vid: this.form.vid,
          startdate: this.form.startdate,
          enddate: this.form.enddate,
        },
        {

```

```

        headers: { "Content-Type": "application/x-www-form-
urlencoded" },
        emulateJSON: true,
    }
)
.then((success) => {
    console.log(success);
    this.$message({ type: "success", message: "添加成功" });
});
}
},
add_driver() {
    this.dialogFormVisible7 = true;
},
add_driver_new() {
    if (this.form.did === "")
        this.$message({ type: "error", message: "身份证号为空！" });
    else if (this.form.dname === "")
        this.$message({ type: "error", message: "姓名为空！" });
    else if (this.form.dphone === "")
        this.$message({ type: "error", message: "电话为空！" });
    else {
        this.$http
        .post(
            main.url + "/driver/add",
            {
                did: this.form.did,
                dname: this.form.dname,
                dsex: this.form.dsex,
                dage: this.form.dage,
                dphone: this.form.dphone,
                ddlicense: this.form.ddlicense,
            },
            {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            }
        )
        .then((success) => {
            console.log(success);
            this.$message({ type: "success", message: "添加成功" });
        });
        this.dialogFormVisible7 = false;
    }
}

```

```

        this.dialogFormVisibleled1 = false;
        this.dialogFormVisibleled1 = true;
    }
},
add_customer() {
    this.dialogFormVisibleled8 = true;
},
add_customer_new() {
    if (this.form.cid === "")
        this.$message({ type: "error", message: "身份证号为空! " });
    else if (this.form.cname === "")
        this.$message({ type: "error", message: "姓名为空! " });
    else if (this.form.cphone === "")
        this.$message({ type: "error", message: "电话为空! " });
    else {
        this.$http
            .post(
                main.url + "/customer/add",
                {
                    cid: this.form.cid,
                    cname: this.form.cname,
                    csex: this.form.csex,
                    cage: this.form.cage,
                    cphone: this.form.cphone,
                    caddress: this.form.caddress,
                },
                {
                    headers: { "Content-Type": "application/x-www-form-urlencoded" },
                    emulateJSON: true,
                }
            )
            .then((success) => {
                console.log(success);
                this.$message({ type: "success", message: "添加成功" });
            });
        this.dialogFormVisibleled8 = false;
        this.dialogFormVisibleled1 = false;
        this.dialogFormVisibleled1 = true;
    }
},
addorder() {
    this.dialogFormVisibleled1 = true;
    this.init_form();
}

```

```

},
addnew() {
    // 添加新收藏
    if (this.form.did === "") this.form.did = "未添加司机";
    if (this.form.cid === "")
        this.$message({ type: "error", message: "客户身份证号为空！" });
    else if (this.form.vid === "")
        this.$message({ type: "error", message: "车牌号为空！" });
    else {
        this.$http
            .post(
                main.url + "/rent_order/add",
                {
                    cid: this.form.cid,
                    vid: this.form.vid,
                    did: this.form.did,
                    model: this.form.model,
                    startdate: this.form.startdate,
                    enddate: this.form.enddate,
                    act_enddate: this.form.act_enddate,
                    adv_pay: this.form.adv_pay,
                    act_pay: this.form.act_pay,
                    bre_pay: this.form.bre_pay,
                    state: this.form.state,
                },
                {
                    headers: { "Content-Type": "application/x-www-form-urlencoded" },
                    emulateJSON: true,
                }
            )
            .then((success) => {
                console.log(success);
                this.$message({ type: "success", message: "添加成功" });
                this.init();
            });
        this.dialogFormVisible1 = false;
    }
},
del_driver(row) {
    // 删除记录
    this.$confirm("请确认是否要删除该记录！", "提示", {
        confirmButtonText: "确定",
        cancelButtonText: "取消",
    });
}

```

```

        type: "warning",
      })
      .then(() => {
        this.$http
          .post(
            main.url + "/rent_order/del",
            { vid: row.vid, cid: row.cid, rid: row.rid },
            {
              headers: {
                "Content-Type": "application/x-www-form-urlencoded",
              },
              emulateJSON: true,
            }
          )
          .then((success) => {
            console.log(success);
            this.$message({ type: "success", message: "已删除" });
            this.init();
          });
      })
      .catch(() => {
        this.$message({ type: "info", message: "已取消" });
      });
    },
  },
};
</script>

```

## 7.9 driver\_admin.vue

```

<template>
  <div>
    <div class="crumbs">
      <el-breadcrumb separator="/">
        <el-breadcrumb-item>
          <i class="el-icon-setting"></i> 管理</el-breadcrumb-item>
        <el-breadcrumb-item>客户列表</el-breadcrumb-item>
      </el-breadcrumb>
    </div>
    <div>
      <el-
table :data="data" border style="width: 90%" ref="multipleTable">

```

```

    <el-table-column
      label="身份证号"
      prop="did"
      width="200px"
    ></el-table-column>
    <el-table-column
      label="姓名"
      prop="dname"
      width="100px"
    ></el-table-column>
    <el-table-column
      label="性别"
      prop="dsex"
      width="60px"
    ></el-table-column>
    <el-table-column
      label="年龄"
      prop="dage"
      width="60px"
    ></el-table-column>
    <el-table-column
      label="电话"
      prop="dphone"
      width="160px"
    ></el-table-column>
    <el-table-column
      label="执照"
      prop="ddlicense"
      width="200px"
    ></el-table-column>
    <el-table-column label="操作" width="320px">
      <template slot-scope="scope" width="100px">
        <el-button type="text" @click="openedit(scope.row)">修改
      </el-button>
        <el-button type="text" @click="del_driver(scope.row)"
          >删除</el-button
        >
      </template>
    </el-table-column> </el-table>
  ><el-button type="primary" @click="add_driver()" align="center"
    >添加司机信息</el-button
  >
  <el-button type="primary" @click="search()" align="center"
    >查询司机信息</el-button

```

```

    ><el-button type="primary" @click="init()" align="center"
    >返回主页</el-button
  >
  <template> </template>
</div>
<el-dialog width="30%" title="修改
" :visible.sync="dialogFormVisible">
  <div>
    <el-form :model="form" :rules="rules" ref="form" label-
width="80px">
      <el-form-item label="ID" prop="did">
        <el-input
          v-model="form.did"
          placeholder="请输入身份证号"
        ></el-input>
      </el-form-item>
      <el-form-item label="姓名" prop="dname">
        <el-input
          v-model="form.dname"
          placeholder="请输入司机姓名"
        ></el-input>
      </el-form-item>
      <el-form-item label="性别" prop="dsex">
        <el-select v-model="form.dsex" placeholder="请设置司机性别">
          <el-option
            v-for="item in options"
            :key="item.value"
            :label="item.label"
            :value="item.value"
          >
          </el-option>
        </el-select>
      </el-form-item>
      <el-form-item label="年龄" prop="dage">
        <el-input
          v-model="form.dage"
          placeholder="请输入司机年龄"
        ></el-input>
      </el-form-item>
      <el-form-item label="电话" prop="dphone">
        <el-input
          v-model="form.dphone"
          placeholder="请输入司机电话"
        ></el-input>

```



```

    </el-form-item>
    <el-form-item label="执照" prop="ddlicense">
      <el-input
        v-model="form.ddlicense"
        placeholder="请输入执照编号"
      ></el-input>
    </el-form-item>
    <el-form-item style="text-align: center">
      <el-button @click="dialogFormVisible = false">取消</el-
button>
      <el-button type="primary" @click="submit('form')">修改</el-
button>
    </el-form-item>
  </el-form>
</div>
</el-dialog>
<el-dialog
  width="30%"
  title="添加司机信息"
  :visible.sync="dialogFormVisible1"
>
  <div>
    <el-form :model="form" :rules="rules" ref="form" label-
width="80px">
      <el-form-item label="ID" prop="did">
        <el-input
          v-model="form.did"
          placeholder="请输入身份证号"
        ></el-input>
      </el-form-item>
      <el-form-item label="姓名" prop="dname">
        <el-input
          v-model="form.dname"
          placeholder="请输入司机姓名"
        ></el-input>
      </el-form-item>
      <el-form-item label="性别" prop="dsex">
        <el-select v-model="form.dsex" placeholder="请设置司机性别">
          <el-option
            v-for="item in options"
            :key="item.value"
            :label="item.label"
            :value="item.value"
          >

```

```

        </el-option>
      </el-select>
    </el-form-item>
    <el-form-item label="年龄" prop="dage">
      <el-input
        v-model="form.dage"
        placeholder="请输入司机年龄"
      ></el-input>
    </el-form-item>
    <el-form-item label="电话" prop="dphone">
      <el-input
        v-model="form.dphone"
        placeholder="请输入司机电话"
      ></el-input>
    </el-form-item>
    <el-form-item label="执照" prop="ddlicense">
      <el-input
        v-model="form.ddlicense"
        placeholder="请输入执照编号"
      ></el-input>
    </el-form-item>
    <el-form-item style="text-align: center">
      <el-button @click="dialogFormVisible1 = false">取消</el-
button>
      <el-button type="primary" @click="addnew(form)">添加</el-
button>
    </el-form-item>
  </el-form>
</div>
</el-dialog>
<el-dialog
  width="30%"
  title="请输入查询信息"
  :visible.sync="dialogFormVisible_search"
>
  <div>
    <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
      <el-form-item label="身份证号" prop="did">
        <el-input
          v-model="form.did"
          placeholder="请输入身份证号"
        ></el-input>
      </el-form-item>

```

```

        <el-form-item style="text-align: center">
            <el-button @click="dialogFormVisibleled_search = false"
            >取消</el-button>
        >
        <el-button type="primary" @click="search_dia(form)">查询
    </el-button>
    </el-form-item>
</el-form>
</div>
</el-dialog>
</div>
</template>

<script>
import main from ".././main";
export default {
  data: function () {
    return {
      data: [],
      options: [
        { value: "男", label: "男" },
        { value: "女", label: "女" },
      ],
      dialogFormVisibleled: false,
      dialogFormVisibleled1: false,
      dialogFormVisibleled_search: false,
      form: {
        did: "",
        dname: "",
        dsex: "",
        dage: 0,
        dphone: "",
        ddlicense: "",
      },
      rules: {
        did: [{ required: true, message: "请输入身份证号", trigger: "blur" }],
        dname: [{ required: true, message: "请输入司机姓名", trigger: "blur" }],
        dphone: [
          { required: true, message: "请输入司机电话", trigger: "blur" },
        ],
      },
    };
  },
};

```

```

    },
    created() {
      if (localStorage.getItem("username") === "") {
        this.$router.replace("/login");
      } else {
        this.init();
      }
    },
    methods: {
      init_form() {
        this.form = {
          did: "",
          dname: "",
          dsex: "",
          dage: 0,
          dphone: "",
          ddlicense: "",
        };
      },
      init() {
        this.$http
          .post(main.url + "/driver/list", {
            headers: { "Content-Type": "application/x-www-form-urlencoded" },
            emulateJSON: true,
          })
          .then((success) => {
            this.data = success.data;
            this.init_form();
          });
      },
      openedit(row) {
        // 修改内容, 按钮按下后
        this.dialogFormVisible = true;
        this.form = {
          did: row.did,
          dname: row.dname,
          dsex: row.dsex,
          dage: row.dage,
          dphone: row.dphone,
          ddlicense: row.ddlicense,
        };
      },
      submit() {

```

```
// 修改内容, 提交
this.$http
  .post(
    main.url + "/driver/update",
    {
      did: this.form.did,
      dname: this.form.dname,
      dsex: this.form.dsex,
      dage: this.form.dage,
      dphone: this.form.dphone,
      ddlicense: this.form.ddlicense,
    },
    {
      headers: { "Content-Type": "application/x-www-form-
urlencoded" },
      emulateJSON: true,
    }
  )
  .then((success) => {
    console.log(success);
    this.$message({ type: "success", message: "修改成功" });
    this.dialogFormVisibleled = false;
    this.init();
  });
},
search() {
  this.dialogFormVisibleled_search = true;
  this.init_form();
},
search_dia() {
  this.$http
    .post(
      main.url + "/driver/search",
      {
        did: this.form.did,
      },
      {
        headers: { "Content-Type": "application/x-www-form-
urlencoded" },
        emulateJSON: true,
      }
    )
    .then((success) => {
      this.data = success.data;
```

```

        this.dialogFormVisibleled_search = false;
        this.init_form();
    });
},
add_driver() {
    this.dialogFormVisibleled1 = true;
    this.init_form();
},
addnew() {
    // 添加新收藏
    if (this.form.did === "")
        this.$message({ type: "error", message: "身份证号为空! " });
    else if (this.form.dname === "")
        this.$message({ type: "error", message: "姓名为空! " });
    else if (this.form.dphone === "")
        this.$message({ type: "error", message: "电话为空! " });
    else {
        this.$http
            .post(
                main.url + "/driver/add",
                {
                    did: this.form.did,
                    dname: this.form.dname,
                    dsex: this.form.dsex,
                    dage: this.form.dage,
                    dphone: this.form.dphone,
                    ddlicense: this.form.ddlicense,
                },
                {
                    headers: { "Content-Type": "application/x-www-form-urlencoded" },
                    emulateJSON: true,
                }
            )
            .then((success) => {
                console.log(success);
                this.$message({ type: "success", message: "添加成功" });
                this.init();
            });
        this.dialogFormVisibleled1 = false;
    }
},
del_driver(row) {
    // 删除记录

```

```

    this.$confirm("请确认是否要删除该记录!", "提示", {
      confirmButtonText: "确定",
      cancelButtonText: "取消",
      type: "warning",
    })
    .then(() => {
      this.$http
        .post(
          main.url + "/driver/del",
          { did: row.did },
          {
            headers: {
              "Content-Type": "application/x-www-form-urlencoded",
            },
            emulateJSON: true,
          }
        )
        .then((success) => {
          console.log(success);
          this.$message({ type: "success", message: "已删除" });
          this.init();
        });
    })
    .catch(() => {
      this.$message({ type: "info", message: "已取消" });
    });
  },
},
};
</script>

```

## 7.10 drive\_admin.vue

```

<template>
  <div>
    <div class="crumbs">
      <el-breadcrumb separator="/">
        <el-breadcrumb-item>
          <i class="el-icon-setting"></i> 管理</el-breadcrumb-item>
        <el-breadcrumb-item>客户列表</el-breadcrumb-item>
      </el-breadcrumb>
    </div>
  </div>

```

```

<div>
  <el-
table :data="data" border style="width: 90%" ref="multipleTable">
    <el-table-column
      label="司机身份证号"
      prop="did"
      width="200px"
    ></el-table-column>
    <el-table-column
      label="车牌号"
      prop="vid"
      width="120px"
    ></el-table-column>
    <el-table-column
      label="开始时间"
      prop="startdate"
      width="120px"
    ></el-table-column>
    <el-table-column
      label="结束时间"
      prop="enddate"
      width="120px"
    ></el-table-column>
    <el-table-column label="操作" width="320px">
      <template slot-scope="scope" width="100px">
        <el-button type="text" @click="openedit(scope.row)">修改
      </el-button>
        <el-button type="text" @click="del_drive(scope.row)"
          >删除</el-button
        >
      </template>
    </el-table-column> </el-table>
    <el-button type="primary" @click="adddrive()" align="center"
    >添加驾车信息</el-button
    >
    <el-button type="primary" @click="search()" align="center"
    >查询驾车信息</el-button
    >
    <el-button type="primary" @click="init()" align="center"
    >返回主页</el-button
    >
  </div>
  <el-dialog width="30%" title="修改
" :visible.sync="dialogFormVisible">

```



```

    <div>
      <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
        <el-form-item label="司机身份证号" prop="did">
          <el-input
            v-model="form.did"
            placeholder="请输入司机身份证号"
          ></el-input>
        </el-form-item>
        <el-form-item label="车牌号" prop="vid">
          <el-input v-model="form.vid" placeholder="请输入车牌号"></el-
input>
        </el-form-item>
        <el-form-item label="开始时间" prop="startdate">
          <el-input
            v-model="form.startdate"
            placeholder="2000-00-00"
          ></el-input>
        </el-form-item>
        <el-form-item label="结束时间" prop="enddate">
          <el-input
            v-model="form.enddate"
            placeholder="2000-00-00"
          ></el-input>
        </el-form-item>
        <el-form-item style="text-align: center">
          <el-button @click="dialogFormVisible = false">取消</el-
button>
          <el-button type="primary" @click="submit('form')">修改</el-
button>
        </el-form-item>
      </el-form>
    </div>
  </el-dialog>
  <el-dialog
    width="30%"
    title="添加客户信息"
    :visible.sync="dialogFormVisible1"
  >
    <div>
      <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
        <el-form-item label="司机身份证号" prop="did">
          <el-input

```

```

        v-model="form.did"
        placeholder="请输入司机身份证号"
    ></el-input>
</el-form-item>
<el-form-item label="车牌号" prop="vid">
    <el-input v-model="form.vid" placeholder="请输入车牌号"></el-
input>
</el-form-item>
<el-form-item label="开始时间" prop="startdate">
    <el-input
        v-model="form.startdate"
        placeholder="2000-00-00"
    ></el-input>
</el-form-item>
<el-form-item label="结束时间" prop="enddate">
    <el-input
        v-model="form.enddate"
        placeholder="2000-00-00"
    ></el-input>
</el-form-item>
<el-form-item style="text-align: center">
    <el-button @click="dialogFormVisibleled1 = false">取消</el-
button>
    <el-button type="primary" @click="addnew(form)">添加</el-
button>
</el-form-item>
</el-form>
</div>
</el-dialog>
<el-dialog
    width="30%"
    title="请输入查询信息"
    :visible.sync="dialogFormVisibleled_search"
>
    <div>
        <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
            <el-form-item label="司机身份证号" prop="did">
                <el-input
                    v-model="form.did"
                    placeholder="请输入司机身份证号"
                ></el-input>
            </el-form-item>
            <el-form-item label="车牌号" prop="vid">

```

```

        <el-input
          v-model="form.vid"
          placeholder="请输入车牌号"
        ></el-input>
      </el-form-item>
      <el-form-item style="text-align: center">
        <el-button @click="dialogFormVisibleed_search = false"
          >取消</el-button>
        <el-button type="primary" @click="search_dia(form)">查询
      </el-button>
    </el-form-item>
  </el-form>
</div>
</el-dialog>
</div>
</template>

<script>
import main from ".././main";
export default {
  data: function () {
    return {
      data: [],
      dialogFormVisibleed: false,
      dialogFormVisibleed1: false,
      dialogFormVisibleed_search: false,
      form: {
        vid: "",
        did: "",
        startdate: "",
        enddate: "",
      },
      rules: {
        vid: [{ required: true, message: "请输入车牌号", trigger: "blur" }],
        did: [
          { required: true, message: "请设置司机身份证号", trigger: "blur" },
        ],
        startdate: [
          { required: true, message: "请输入租赁日期", trigger: "blur" },
        ],
        enddate: [

```

```

        { required: true, message: "请输入归还日期", trigger: "blur" },
      ],
    },
  };
},
created() {
  if (localStorage.getItem("username") === "") {
    this.$router.replace("/login");
  } else {
    this.init();
  }
},
methods: {
  init_form() {
    this.form = {
      vid: "",
      did: "",
      startdate: "",
      enddate: "",
    };
  },
  init() {
    this.$http
      .post(main.url + "/drive/list", {
        headers: { "Content-Type": "application/x-www-form-urlencoded" },
        emulateJSON: true,
      })
      .then((success) => {
        this.data = success.data;
        this.init_form();
      });
  },
  openedit(row) {
    // 修改内容, 按钮按下后
    this.dialogFormVisible = true;
    this.form = {
      vid: row.vid,
      did: row.did,
      startdate: row.startdate,
      enddate: row.enddate,
    };
  },
  submit() {

```

```
// 修改内容, 提交
this.$http
  .post(
    main.url + "/drive/update",
    {
      vid: this.form.vid,
      did: this.form.did,
      startdate: this.form.startdate,
      enddate: this.form.enddate,
    },
    {
      headers: { "Content-Type": "application/x-www-form-urlencoded" },
      emulateJSON: true,
    }
  )
  .then((success) => {
    console.log(success);
    this.$message({ type: "success", message: "修改成功" });
    this.dialogFormVisible = false;
    this.init();
  });
},
search() {
  this.dialogFormVisible_search = true;
  this.init_form();
},
search_dia() {
  this.$http
    .post(
      main.url + "/drive/search",
      {
        vid: this.form.vid,
        did: this.form.did,
      },
      {
        headers: { "Content-Type": "application/x-www-form-urlencoded" },
        emulateJSON: true,
      }
    )
    .then((success) => {
      this.data = success.data;
      this.dialogFormVisible_search = false;
    });
}
```

```

        this.init_form();
    });
},
adddrive() {
    this.dialogFormVisible1 = true;
    this.init_form();
},
addnew() {
    // 添加新收藏
    if (this.form.vid === "")
        this.$message({ type: "error", message: "车牌号为空! " });
    else if (this.form.did === "")
        this.$message({ type: "error", message: "司机身份证号为空! " });
    else if (this.form.startdate === "")
        this.$message({ type: "error", message: "租赁日期为空! " });
    else if (this.form.enddate === "")
        this.$message({ type: "error", message: "归还日期为空! " });
    else {
        this.$http
            .post(
                main.url + "/drive/add",
                {
                    vid: this.form.vid,
                    did: this.form.did,
                    startdate: this.form.startdate,
                    enddate: this.form.enddate,
                },
                {
                    headers: { "Content-Type": "application/x-www-form-urlencoded" },
                    emulateJSON: true,
                }
            )
            .then((success) => {
                console.log(success);
                this.$message({ type: "success", message: "添加成功" });
                this.form = {
                    vid: "",
                    did: "",
                    startdate: "",
                    enddate: "",
                };
                this.init();
            });
    });
}

```

```

        this.dialogFormVisible1 = false;
    }
},
del_drive(row) {
    // 删除记录
    this.$confirm("请确认是否要删除该记录!", "提示", {
        confirmButtonText: "确定",
        cancelButtonText: "取消",
        type: "warning",
    })
    .then(() => {
        this.$http
            .post(
                main.url + "/drive/del",
                { did: row.did, vid: row.vid },
                {
                    headers: {
                        "Content-Type": "application/x-www-form-urlencoded",
                    },
                    emulateJSON: true,
                }
            )
            .then((success) => {
                console.log(success);
                this.$message({ type: "success", message: "已删除" });
                this.init();
            });
    })
    .catch(() => {
        this.$message({ type: "info", message: "已取消" });
    });
},
},
};
</script>

```

## 7.11 customer\_admin.vue

```

<template>
  <div>
    <div class="crumbs">
      <el-breadcrumb separator="/">
        <el-breadcrumb-item

```

```

        <i class="el-icon-setting"></i> 管理</el-breadcrumb-item>
      </el-breadcrumb-item>客户列表</el-breadcrumb-item>
    </el-breadcrumb>
  </div>
  <div>
    <el-
table :data="data" border style="width: 90%" ref="multipleTable">
      <el-table-column
        label="身份证号"
        prop="cid"
        width="200px"
      ></el-table-column>
      <el-table-column
        label="姓名"
        prop="cname"
        width="100px"
      ></el-table-column>
      <el-table-column
        label="性别"
        prop="csex"
        width="60px"
      ></el-table-column>
      <el-table-column
        label="年龄"
        prop="cage"
        width="60px"
      ></el-table-column>
      <el-table-column
        label="电话"
        prop="cphone"
        width="160px"
      ></el-table-column>
      <el-table-column
        label="地址"
        prop="caddress"
        width="200px"
      ></el-table-column>
      <el-table-column label="操作" width="320px">
        <template slot-scope="scope" width="100px">
          <el-button type="text" @click="openedit(scope.row)">修改
</el-button>
          <el-button type="text" @click="del_cus(scope.row)">删除</el-
button>

```



```

        </template>
      </el-table-column> </el-table>
    <el-button type="primary" @click="add_cus()" align="center"
      >添加客户信息</el-button>
    <
    <el-button type="primary" @click="search()" align="center"
      >查询客户信息</el-button>
    <el-button type="primary" @click="init()" align="center"
      >返回主页</el-button>
    >
  </template> </template>
</div>
<el-dialog width="30%" title="修改
" :visible.sync="dialogFormVisible">
  <div>
    <el-form :model="form" :rules="rules" ref="form" label-
width="80px">
      <el-form-item label="ID" prop="cid">
        <el-input
          v-model="form.cid"
          placeholder="请输入身份证号"
        ></el-input>
      </el-form-item>
      <el-form-item label="姓名" prop="cname">
        <el-input
          v-model="form.cname"
          placeholder="请输入客户姓名"
        ></el-input>
      </el-form-item>
      <el-form-item label="性别" prop="csex">
        <el-select v-model="form.csex" placeholder="请设置客户性别">
          <el-option
            v-for="item in options"
            :key="item.value"
            :label="item.label"
            :value="item.value"
          >
          </el-option>
        </el-select>
      </el-form-item>
      <el-form-item label="年龄" prop="cage">
        <el-input
          v-model="form.cage"
          placeholder="请输入客户年龄"

```

```

        ></el-input>
    </el-form-item>
    <el-form-item label="电话" prop="cphone">
        <el-input
            v-model="form.cphone"
            placeholder="请输入客户电话"
        ></el-input>
    </el-form-item>
    <el-form-item label="地址" prop="caddress">
        <el-input
            v-model="form.caddress"
            placeholder="请输入客户地址"
        ></el-input>
    </el-form-item>
    <el-form-item style="text-align: center">
        <el-button @click="dialogFormVisible = false">取消</el-
button>
        <el-button type="primary" @click="submit('form')">修改</el-
button>
    </el-form-item>
</el-form>
</div>
</el-dialog>
<el-dialog
    width="30%"
    title="添加客户信息"
    :visible.sync="dialogFormVisible1"
>
    <div>
        <el-form :model="form" :rules="rules" ref="form" label-
width="80px">
            <el-form-item label="ID" prop="cid">
                <el-input
                    v-model="form.cid"
                    placeholder="请输入身份证号"
                ></el-input>
            </el-form-item>
            <el-form-item label="姓名" prop="cname">
                <el-input
                    v-model="form.cname"
                    placeholder="请输入客户姓名"
                ></el-input>
            </el-form-item>
            <el-form-item label="性别" prop="csex">

```

```

        <el-select v-model="form.csex" placeholder="请设置客户性别">
          <el-option
            v-for="item in options"
            :key="item.value"
            :label="item.label"
            :value="item.value"
          >
        </el-option>
      </el-select>
    </el-form-item>
    <el-form-item label="年龄" prop="cage">
      <el-input
        v-model="form.cage"
        placeholder="请输入客户年龄"
      ></el-input>
    </el-form-item>
    <el-form-item label="电话" prop="cphone">
      <el-input
        v-model="form.cphone"
        placeholder="请输入客户电话"
      ></el-input>
    </el-form-item>
    <el-form-item label="地址" prop="caddress">
      <el-input
        v-model="form.caddress"
        placeholder="请输入客户地址"
      ></el-input>
    </el-form-item>
    <el-form-item style="text-align: center">
      <el-button @click="dialogFormVisible1 = false">取消</el-
button>
      <el-button type="primary" @click="addnew(form)">添加</el-
button>
    </el-form-item>
  </el-form>
</div>
</el-dialog>
<el-dialog
  width="30%"
  title="请输入查询信息"
  :visible.sync="dialogFormVisible_search"
>
  <div>

```

```

    <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
      <el-form-item label="客户身份证号" prop="cid">
        <el-input
          v-model="form.cid"
          placeholder="请输入客户身份证号"
        ></el-input>
      </el-form-item>
      <el-form-item style="text-align: center">
        <el-button @click="dialogFormVisibleled_search = false"
          >取消</el-button>
        <el-button type="primary" @click="search_dia(form)">查询
      </el-button>
    </el-form-item>
  </el-form>
</div>
</el-dialog>
</div>
</template>

<script>
import main from "../../main";
export default {
  data: function () {
    return {
      data: [],
      options: [
        { value: "男", label: "男" },
        { value: "女", label: "女" },
      ],
      dialogFormVisibleled: false,
      dialogFormVisibleled1: false,
      dialogFormVisibleled_search: false,
      form: {
        cid: "",
        cname: "",
        csex: "",
        cage: 0,
        cphone: "",
        caddress: "",
      },
      rules: {

```

```

        cid: [{ required: true, message: "请输入身份证号", trigger: "blur" }],
        cname: [{ required: true, message: "请输入客户姓名", trigger: "blur" }],
        cphone: [
            { required: true, message: "请输入客户电话", trigger: "blur" },
        ],
    },
};
},
created() {
    if (localStorage.getItem("username") === "") {
        this.$router.replace("/login");
    } else {
        this.init();
    }
},
methods: {
    init_form() {
        this.form = {
            cid: "",
            cname: "",
            csex: "",
            cage: 0,
            cphone: "",
            caddress: "",
        };
    },
    init() {
        this.$http
            .post(main.url + "/customer/list", {
                headers: { "Content-Type": "application/x-www-form-urlencoded" },
                emulateJSON: true,
            })
            .then((success) => {
                this.data = success.data;
                this.init_form();
            });
    },
    openedit(row) {
        // 修改内容, 按钮按下后
        this.dialogFormVisible = true;
        this.form = {

```

```

        cid: row.cid,
        cname: row.cname,
        csex: row.csex,
        cage: row.cage,
        cphone: row.cphone,
        caddress: row.caddress,
    };
},
submit() {
    // 修改内容, 提交
    this.$http
        .post(
            main.url + "/customer/update",
            {
                cid: this.form.cid,
                cname: this.form.cname,
                csex: this.form.csex,
                cage: this.form.cage,
                cphone: this.form.cphone,
                caddress: this.form.caddress,
            },
            {
                headers: { "Content-Type": "application/x-www-form-urlencoded" },
                emulateJSON: true,
            }
        )
        .then((success) => {
            console.log(success);
            this.$message({ type: "success", message: "修改成功" });
            this.dialogFormVisibleled = false;
            this.init();
        });
},
search() {
    this.dialogFormVisibleled_search = true;
    this.init_form();
},
search_dia() {
    this.$http
        .post(
            main.url + "/customer/search",
            {
                cid: this.form.cid,

```

```

    },
    {
        headers: { "Content-Type": "application/x-www-form-
urlencoded" },
        emulateJSON: true,
    }
)
.then((success) => {
    this.data = success.data;
    this.dialogFormVisibleled_search = false;
    this.init_form();
});
},
add_cus() {
    this.dialogFormVisibleled1 = true;
    this.init_form();
},
addnew() {
    if (this.form.cid === "")
        this.$message({ type: "error", message: "身份证号为空! " });
    else if (this.form.cname === "")
        this.$message({ type: "error", message: "姓名为空! " });
    else if (this.form.cphone === "")
        this.$message({ type: "error", message: "电话为空! " });
    else {
        this.$http
        .post(
            main.url + "/customer/add",
            {
                cid: this.form.cid,
                cname: this.form.cname,
                csex: this.form.csex,
                cage: this.form.cage,
                cphone: this.form.cphone,
                caddress: this.form.caddress,
            },
            {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            }
        )
        .then((success) => {
            console.log(success);

```

```

        this.$message({ type: "success", message: "添加成功" });
        this.dialogFormVisible1 = false;
        this.init();
    });
}
},
del_cus(row) {
    // 删除记录
    this.$confirm("请确认是否要删除该记录!", "提示", {
        confirmButtonText: "确定",
        cancelButtonText: "取消",
        type: "warning",
    })
    .then(() => {
        this.$http
            .post(
                main.url + "/customer/del",
                { cid: row.cid },
                {
                    headers: {
                        "Content-Type": "application/x-www-form-urlencoded",
                    },
                    emulateJSON: true,
                }
            )
            .then((success) => {
                console.log(success);
                this.$message({ type: "success", message: "已删除" });
                this.init();
            });
    })
    .catch(() => {
        this.$message({ type: "info", message: "已取消" });
    });
},
},
};
</script>

```

## 7.12 account\_admin.vue

```

<template>
  <div>

```



```

<div class="crumbs">
  <el-breadcrumb separator="/">
    <el-breadcrumb-item
      ><i class="el-icon-setting"></i> 管理</el-breadcrumb-item
    >
    <el-breadcrumb-item>用户列表</el-breadcrumb-item>
  </el-breadcrumb>
</div>
<div>
  <el-
table :data="data" border style="width: 90%" ref="multipleTable">
  <el-table-column label="ID" prop="id" width="80px"></el-table-
column>
  <el-table-column
    label="用户名"
    prop="username"
    width="200px"
  ></el-table-column>
  <el-table-column
    label="角色"
    prop="role"
    width="150px"
  ></el-table-column>
  <el-table-column
    label="创建时间"
    prop="ctime"
    width="250px"
  ></el-table-column>
  <el-table-column label="操作" width="200px">
    <template slot-scope="scope" width="100px">
      <el-button type="text" @click="openedit(scope.row)">修改
</el-button>
      <el-button type="text" @click="del_user(scope.row)">删除
</el-button>
    </template>
  </el-table-column>
</el-table>
  <el-button type="primary" @click="adduser()" align="center"
  >添加账号</el-button
  >
  <template> </template>
</div>
  <el-dialog width="30%" title="修改账号
" :visible.sync="dialogFormVisible">

```

```

<div>
  <el-form :model="form" :rules="rules" ref="form" label-
width="150px">
    <el-form-item label="新密码" prop="password1">
      <el-input
        v-model="form.password1"
        type="password"
        placeholder="请输入新密码"
      ></el-input>
    </el-form-item>
    <el-form-item label="确认新密码" prop="password2">
      <el-input
        v-model="form.password2"
        type="password"
        placeholder="请确认新密码"
      ></el-input>
    </el-form-item>
    <el-form-item style="text-align: center">
      <el-button type="primary" @click="change_pass()"
        >修改密码</el-button
      >
    </el-form-item>
    <el-form-item label="账号类型" prop="role">
      <el-select v-model="form.role" placeholder="请选择账号类型">
        <el-option
          v-for="item in options"
          :key="item.value"
          :label="item.label"
          :value="item.value"
        >
        </el-option>
      </el-select>
    </el-form-item>
    <el-form-item style="text-align: center">
      <el-button type="primary" @click="change_role()"
        >修改账号类型</el-button
      >
    </el-form-item>
    <el-form-item style="text-align: center">
      <el-button @click="dialogFormVisible = false">取 消</el-
button>
    </el-form-item>
  </el-form>
</div>

```

```

</el-dialog>
<el-dialog
  width="30%"
  title="添加账号"
  :visible.sync="dialogFormVisible1"
>
  <div>
    <el-form :model="form" :rules="rules1" ref="form" label-
width="150px">
      <el-form-item label="用户名" prop="username">
        <el-input
          v-model="form.username"
          placeholder="请输入用户名"
        ></el-input>
      </el-form-item>
      <el-form-item label="密码" prop="password1">
        <el-input
          v-model="form.password1"
          type="password"
          placeholder="请输入密码"
        ></el-input>
      </el-form-item>
      <el-form-item label="确认密码" prop="password2">
        <el-input
          v-model="form.password2"
          type="password"
          placeholder="请确认密码"
        ></el-input>
      </el-form-item>
      <el-form-item label="账号类型" prop="role">
        <el-select v-model="form.role" placeholder="请选择账号类型">
          <el-option
            v-for="item in options"
            :key="item.value"
            :label="item.label"
            :value="item.value"
          >
            </el-option>
        </el-select>
      </el-form-item>
      <el-form-item style="text-align: center">
        <el-button @click="dialogFormVisible1 = false">取消</el-
button>

```

```

        <el-button type="primary" @click="adduser_submit()">添加
    </el-button>
    </el-form-item>
  </el-form>
</div>
</el-dialog>
</div>
</template>

<script>
import main from ".././main";
export default {
  data: function () {
    var checkpasswd = (rule, value, callback) => {
      if (value === "") {
        callback(new Error("请再次确认新密码"));
      } else if (value !== this.form.password1) {
        callback(new Error("两次输入的新密码不一致!"));
      } else {
        callback();
      }
    };
    var checkuser = (rule, value, callback) => {
      for (let i = 0; i < this.data.length; i++) {
        if (this.data[i].username === value) {
          this.form.ban = "1";
          callback(new Error("该用户名已存在"));
        }
      }
    };
    return {
      data: [
        { id: 1, username: "haha", role: "普通用户", ctime: "2010-11-12" },
      ],
      dialogFormVisible: false,
      dialogFormVisible1: false,
      options: [
        { value: 0, label: "普通用户" },
        { value: 1, label: "管理员" },
      ],
      form: {
        id: "",
        username: "",

```

```

        password1: "",
        password2: "",
        role: 0,
        ban: "0", // 重名检测, 用户名是否已经存在, 0 存在 1 不存在
    },
    rules: {
        password1: [
            { required: true, message: "请输入新密码", trigger: "blur" },
        ],
        password2: [
            { required: true, message: "请确认新密码", trigger: "blur" },
            { validator: checkpasswd, trigger: "blur" },
        ],
        role: [{ required: true, message: "请选择账号类型", trigger: "blur" }],
    },
    rules1: {
        username: [
            { required: true, message: "请输入用户名", trigger: "blur" },
            { validator: checkuser, trigger: "blur" },
        ],
        password1: [{ required: true, message: "请输入密码", trigger: "blur" }],
        password2: [
            { required: true, message: "请确认密码", trigger: "blur" },
            { validator: checkpasswd, trigger: "blur" },
        ],
        role: [{ required: true, message: "请选择账号类型", trigger: "blur" }],
    },
};

created() {
    if (localStorage.getItem("username") === "") {
        this.$router.replace("/login");
    } else if (localStorage.getItem("role") !== 1) {
        this.$router.replace("/admin");
    } else {
        this.init();
    }
},

methods: {
    init() {
        this.$http.get(main.url + "/login/list").then((success) => {

```

```

        console.log(success.data);
        this.data = [
            { id: 1, username: "haha", role: "普通用户", ctime: "2010-11-
12" },
        ];
        this.data = success.data;
        for (let i = 0; i < this.data.length; i++) {
            if (this.data[i].role === 0) this.data[i].role = "普通用户";
            else this.data[i].role = "管理员";
        }
    });
},
openedit(row) {
    this.dialogFormVisible = true;
    this.form = {
        password1: "",
        password2: "",
        id: row.id,
        role: 0,
    };
},
change_pass() {
    // 修改密码
    if (this.form.password1 !== this.form.password2) {
        this.$message({ type: "error", message: "两次密码不一样! " });
    } else {
        let crypto = require("crypto");
        const md5 = crypto.createHash("md5");
        md5.update(this.form.password1);
        let md5password = md5.digest("hex");
        this.$http
            .post(
                main.url + "/login/update",
                {
                    id: this.form.id,
                    password: md5password,
                },
                {
                    headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                    emulateJSON: true,
                }
            )
            .then((success) => {

```

```
        console.log(success.data);
        this.$message({ type: "success", message: "修改成功" });
        this.form = {
            id: "",
            username: "",
            password1: "",
            password2: "",
            role: 0,
            ban: "0",
        };
        this.dialogFormVisible = false;
    });
}
},
adduser() {
    this.form = {
        id: "",
        username: "",
        password1: "",
        password2: "",
        role: 0,
        ban: "0",
    };
    this.dialogFormVisible1 = true;
    this.form = {
        id: "",
        username: "",
        password1: "",
        password2: "",
        role: 0,
        ban: "0",
    };
},
adduser_submit() {
    // 添加新用户
    if (this.form.username === "")
        this.$message({ type: "error", message: "用户名不能为空！" });
    else if (this.form.password1 !== this.form.password2)
        this.$message({ type: "error", message: "两次密码不一样！" });
    else if (this.form.ban === "1")
        this.$message({ type: "error", message: "用户名已存在！" });
    else {
        let crypto = require("crypto");
        const md5 = crypto.createHash("md5");
```

```

md5.update(this.form.password1);
let md5password = md5.digest("hex");
this.$http
  .post(
    main.url + "/login/add",
    {
      username: this.form.username,
      password: md5password,
      role: this.form.role,
    },
    {
      headers: { "Content-Type": "application/x-www-form-urlencoded" },
      emulateJSON: true,
    }
  )
  .then((success) => {
    console.log(success.data);
    this.$message({ type: "success", message: "添加成功" });
    this.form = {
      id: "",
      username: "",
      password1: "",
      password2: "",
      role: 0,
      ban: "0",
    };
    this.init();
  });
this.dialogFormVisible1 = false;
},
del_user(row) {
  // 删除用户
  if (row.username === localStorage.getItem("username"))
    this.$message({ type: "info", message: "不能删除自己!" });
  else {
    this.$confirm("请确认是否要删除该用户!", "提示", {
      confirmButtonText: "确定",
      cancelButtonText: "取消",
      type: "warning",
    })
    .then(() => {
      this.$http

```



```

        .post(
            main.url + "/login/del",
            { id: row.id },
            {
                headers: {
                    "Content-Type": "application/x-www-form-
urlencoded",
                },
                emulateJSON: true,
            }
        )
        .then((success) => {
            console.log(success.data);
            this.$message({ type: "success", message: "已删除" });
            this.form = {
                id: "",
                username: "",
                password1: "",
                password2: "",
                role: 0,
                ban: "0",
            };
            this.init();
        });
    })
    .catch(() => {
        this.$message({ type: "info", message: "已取消" });
    });
}
},
change_role() {
    // 修改账号类型
    this.$http
        .post(
            main.url + "/login/update_role",
            {
                id: this.form.id,
                role: this.form.role,
            },
            {
                headers: { "Content-Type": "application/x-www-form-
urlencoded" },
                emulateJSON: true,
            }
        )

```

```

    )
    .then((success) => {
      console.log(success.data);
      this.$message({ type: "success", message: "修改成功" });
      this.form = {
        id: "",
        username: "",
        password1: "",
        password2: "",
        role: 0,
        ban: "0",
      };
      this.dialogFormVisible = false;
    });
    this.init();
  },
},
};
</script>

```

## 7.13 Sidebar.vue

```

<template>
  <div class="sidebar">
    <el-menu :default-active="onRoutes" class="el-menu-vertical-
demo" theme="dark" unique-opened router>
      <template v-for="item in items">
        <template v-if="item.subs">
          <el-submenu :key="item.label" :index="item.index">
            <template slot="title"><i :class="item.icon"></i>{{ item.title }}</template>
            <el-menu-item v-
for="(subItem,i) in item.subs" :key="i" :index="subItem.index">{{ subIt
em.title }}

          </el-menu-item>
        </el-submenu>
      </template>
      <template v-else>
        <el-menu-
item :key="item.label" :index="item.index">
          <i :class="item.icon"></i>{{ item.title }}
        </el-menu-item>
      </template>
    </el-menu>
  </div>
</template>

```

```
        </template>
      </el-menu>
    </div>
  </template>

<script>
  export default {
    data() {
      return {
        items:
          [
            {
              icon: 'el-icon-setting',
              index: 'account_admin',
              title: '用户管理',
            },
            {
              icon: 'el-icon-setting',
              index: 'vehicle_admin',
              title: '车辆管理',
            },
            {
              icon: 'el-icon-setting',
              index: 'customer_admin',
              title: '客户管理',
            },
            {
              icon: 'el-icon-setting',
              index: 'driver_admin',
              title: '司机管理',
            },
            {
              icon: 'el-icon-setting',
              index: 'drive_admin',
              title: '驾驶信息管理',
            },
            {
              icon: 'el-icon-setting',
              index: 'rent_order_admin',
              title: '订单管理',
            },
          ],
      }
    },
  },
}
```

```

        created(){
            if(localStorage.getItem('username')==="){
                this.$router.replace('/login');
            }
        },
        computed:{
            onRoutes(){
                return this.$route.path.replace('/', '');
            }
        }
    }
}
</script>

<style scoped>
    .sidebar{
        display: block;
        position: absolute;
        width: 250px;
        left: 0;
        top: 70px;
        bottom:0;
        background: #2E363F;
    }
    .sidebar > ul {
        height:100%;
    }
</style>

```

## 7.14 Home. vue

```

<template>
    <div class="wrapper">
        <v-head></v-head>
        <v-sidebar></v-sidebar>
        <div class="content">
            <transition name="move" mode="out-in"><router-view></router-view></transition>
        </div>
    </div>
</template>

<script>
    import vHead from './Header.vue';

```

```

import vSidebar from './Sidebar.vue';
export default {
  components:{
    vHead, vSidebar
  }
}
</script>

```

## 7.15 Header. vue

```

<template>
  <div class="header">
    <div class="logo">车辆租赁管理系统</div>
    <div class="user-info">
      <el-dropdown trigger="click" @command="handleCommand">
        <span class="el-dropdown-link">
          
          {{username}}
        </span>
        <el-dropdown-menu slot="dropdown">
          <el-dropdown-item command="change_pass">修改密码
</el-dropdown-item>
          <el-dropdown-item command="logout">退出</el-
dropdown-item>
        </el-dropdown-menu>
      </el-dropdown>
    </div>
    <el-dialog
      width="30%"
      title="修改密码"
      :visible.sync="dialogFormVisible1">
      <div>
        <el-
form :model="form1" :rules="rules1" ref="form1" label-width="150px">
          <el-form-item label="新密码" prop="name">
            <el-input v-
model="password1" type="password" placeholder="请输入新密码"></el-input>
          </el-form-item>
          <el-form-item label="确认密码" prop="name">
            <el-input v-
model="password2" type="password" placeholder="请输入确认密码"></el-
input>

```

```

        </el-form-item>
        <el-form-item style="text-align: center" >
            <el-
button @click="dialogFormVisible1 = false">取消</el-button>
            <el-button type="primary" @click="submit()">确认
</el-button>
        </el-form-item>
    </el-form>
</div>
</el-dialog>
</div>
</template>
<script>
import main from ".././main";
export default {
    data() {
        return {
            name: '用户名',
            dialogFormVisible1:false,
            password1:'',
            password2:'',
            form1:{},
        }
    },
    computed:{
        username(){
            let username = localStorage.getItem('username');
            return username ? username : this.name;
        }
    },
    methods:{
        handleCommand(command) {
            if(command == 'loginout'){
                localStorage.removeItem('username');
                localStorage.removeItem('role');
                localStorage.removeItem('id');
                this.$router.push('/');
            }
            else if(command == 'change_pass'){this.dialogFormVisibl
eed1 = true}
        },
        submit(){
            if(this.password1=="")

```

```

        this.$message({type: 'info', message: '密码不能为空!
    });

    else if(this.password1!==this.password2)
        this.$message({type: 'info', message: '确认密码与新密
码不一致! '});
    else{
        let crypto = require('crypto');
        const md5 = crypto.createHash('md5');
        md5.update(this.password1);
        let md5password = md5.digest('hex');
        this.$http.post(main.url+"/login/update",
            {
                'id': localStorage.getItem('id'),
                'password': md5password,
            },
            {
                headers: {'Content-Type': 'application/x-
www-form-urlencoded'},
                emulateJSON: true
            }).then(
                /*success=>{this.$message({type: 'success', mes
sage: '密码修改成功! '})}*/
            );
        this.dialogFormVisible1 = false;
    }
}
}
}
</script>
<style scoped>
    .header {
        position: relative;
        box-sizing: border-box;
        width: 100%;
        height: 70px;
        font-size: 22px;
        line-height: 70px;
        color: #fff;
    }
    .header .logo{
        float: left;
        width: 250px;
        text-align: center;
    }

```

```
.user-info {
    float: right;
    padding-right: 50px;
    font-size: 16px;
    color: #fff;
}
.user-info .el-dropdown-link{
    position: relative;
    display: inline-block;
    padding-left: 50px;
    color: #fff;
    cursor: pointer;
    vertical-align: middle;
}
.user-info .user-logo{
    position: absolute;
    left:0;
    top:15px;
    width:40px;
    height:40px;
    border-radius: 50%;
}
.el-dropdown-menu__item{
    text-align: center;
}
</style>
```

## MySQL 数据库操作

```
/*
Navicat Premium Data Transfer

Source Server        : mysql
Source Server Type   : MySQL
Source Server Version : 80025
Source Host          : localhost:3306
Source Schema        : data_1

Target Server Type    : MySQL
Target Server Version : 80025
File Encoding        : 65001
```



Date: 15/07/2021 20:11:17

\*/

SET NAMES utf8mb4;

SET FOREIGN\_KEY\_CHECKS = 0;

-----  
-- Table structure for customer  
-----

DROP TABLE IF EXISTS `customer`;

CREATE TABLE `customer` (

  `cid` char(50) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NOT NULL,

  `cname` varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NOT NULL,

  `csex` varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NOT NULL,

  `cage` int NOT NULL,

  `cphone` varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NULL DEFAULT NULL,

  `caddress` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NULL DEFAULT NULL,

  PRIMARY KEY (`cid`) USING BTREE

) ENGINE = InnoDB CHARACTER SET = utf8mb4 COLLATE = utf8mb4\_0900\_ai\_ci ROW\_FORMAT = Dynamic;

-----  
-- Table structure for drive  
-----

DROP TABLE IF EXISTS `drive`;

CREATE TABLE `drive` (

  `vid` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NOT NULL,

  `did` char(50) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NOT NULL,

  `startdate` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NULL DEFAULT NULL,

  `enddate` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NULL DEFAULT NULL,

  PRIMARY KEY (`vid`, `did`) USING BTREE,

  INDEX `drive\_did` (`did`) USING BTREE,

  CONSTRAINT `drive\_ibfk\_1` FOREIGN KEY (`did`) REFERENCES `driver` (`did`) ON DELETE RESTRICT ON UPDATE RESTRICT,

```

        CONSTRAINT `drive_ibfk_2` FOREIGN KEY (`vid`) REFERENCES `vehicle`
        (`vid`) ON DELETE RESTRICT ON UPDATE RESTRICT
    ) ENGINE = InnoDB CHARACTER SET = utf8mb4 COLLATE = utf8mb4_0900_ai_ci
    ROW_FORMAT = Dynamic;

-- -----
-- Table structure for driver
-- -----
DROP TABLE IF EXISTS `driver`;
CREATE TABLE `driver` (
    `did` char(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT
    NULL,
    `dname` varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
    NULL DEFAULT NULL,
    `dsex` varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
    NULL DEFAULT NULL,
    `dage` int NULL DEFAULT NULL,
    `dphone` varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
    NULL DEFAULT NULL,
    `ddlicense` varchar(255) CHARACTER SET utf8mb4 COLLATE
    utf8mb4_0900_ai_ci NULL DEFAULT NULL,
    PRIMARY KEY (`did`) USING BTREE
) ENGINE = InnoDB CHARACTER SET = utf8mb4 COLLATE = utf8mb4_0900_ai_ci
ROW_FORMAT = Dynamic;

-- -----
-- Table structure for login
-- -----
DROP TABLE IF EXISTS `login`;
CREATE TABLE `login` (
    `id` int NOT NULL AUTO_INCREMENT,
    `username` varchar(100) CHARACTER SET utf8mb4 COLLATE
    utf8mb4_german2_ci NOT NULL,
    `password` varchar(255) CHARACTER SET utf8mb4 COLLATE
    utf8mb4_german2_ci NOT NULL,
    `role` tinyint UNSIGNED NULL DEFAULT 0,
    `ctime` datetime NULL DEFAULT CURRENT_TIMESTAMP,
    PRIMARY KEY (`id`) USING BTREE
) ENGINE = InnoDB AUTO_INCREMENT = 11 CHARACTER SET = utf8mb4 COLLATE =
    utf8mb4_german2_ci ROW_FORMAT = Dynamic;

-- -----
-- Table structure for rent_order
-- -----

```

```

DROP TABLE IF EXISTS `rent_order`;
CREATE TABLE `rent_order` (
  `vid` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
NOT NULL,
  `cid` char(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT
NULL,
  `startdate` varchar(255) CHARACTER SET utf8mb4 COLLATE
utf8mb4_german2_ci NULL DEFAULT NULL,
  `enddate` varchar(255) CHARACTER SET utf8mb4 COLLATE
utf8mb4_german2_ci NULL DEFAULT NULL,
  `model` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
NULL DEFAULT NULL,
  `adv_pay` int NULL DEFAULT 0,
  `act_pay` int NULL DEFAULT 0,
  `did` char(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NULL
DEFAULT NULL,
  `rid` int(10) UNSIGNED ZEROFILL NOT NULL AUTO_INCREMENT,
  `bre_pay` int NULL DEFAULT 0,
  `state` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
NULL DEFAULT NULL,
  `act_enddate` varchar(255) CHARACTER SET utf8mb4 COLLATE
utf8mb4_german2_ci NULL DEFAULT NULL,
  PRIMARY KEY (`rid`) USING BTREE,
  INDEX `vid_ord` (`vid`) USING BTREE,
  INDEX `cid_ord` (`cid`) USING BTREE,
  INDEX `did_ord` (`did`) USING BTREE,
  CONSTRAINT `cid_ord` FOREIGN KEY (`cid`) REFERENCES `customer` (`cid`)
ON DELETE RESTRICT ON UPDATE RESTRICT,
  CONSTRAINT `did_ord` FOREIGN KEY (`did`) REFERENCES `driver` (`did`)
ON DELETE RESTRICT ON UPDATE RESTRICT,
  CONSTRAINT `vid_ord` FOREIGN KEY (`vid`) REFERENCES `vehicle` (`vid`)
ON DELETE RESTRICT ON UPDATE RESTRICT
) ENGINE = InnoDB AUTO_INCREMENT = 20 CHARACTER SET = utf8mb4 COLLATE =
utf8mb4_german2_ci ROW_FORMAT = Dynamic;

-- -----
-- Table structure for vehicle
-- -----

DROP TABLE IF EXISTS `vehicle`;
CREATE TABLE `vehicle` (
  `vid` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
NOT NULL,
  `vtype` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
NULL DEFAULT NULL,

```

```
`vbrand` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
NULL DEFAULT NULL,
`vrent` float NULL DEFAULT NULL,
`vcondition` varchar(255) CHARACTER SET utf8mb4 COLLATE
utf8mb4_0900_ai_ci NULL DEFAULT NULL,
`vdel` int NULL DEFAULT 0,
`vbook` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
NULL DEFAULT '0',
PRIMARY KEY (`vid`) USING BTREE
) ENGINE = InnoDB CHARACTER SET = utf8mb4 COLLATE = utf8mb4_0900_ai_ci
ROW_FORMAT = Dynamic;

SET FOREIGN_KEY_CHECKS = 1;
```