数据库名称为 car\_rent

会员信息表 t\_user

会员编号 uid 会员名字 uname 入会时间 udate 登陆密码 upsw

CREATE TABLE t\_user (

uid VARCHAR(45) NOT NULL,

uname VARCHAR(45) NOT NULL,

udate DATE NOT NULL,

upsw VARCHAR(45) NOT NULL,

PRIMARY KEY (uid));

车辆信息表 t\_car\_info

车辆编号 carid 车型名称 cartype 车辆牌照 carlic

CREATE TABLE t\_car\_info (

carid VARCHAR(45) NOT NULL,

cartype VARCHAR(45) NOT NULL,

carlic VARCHAR(45) NOT NULL,

PRIMARY KEY (carid));

车型信息表 t\_type\_info

车型名称 cartype 车型价格 rentprice 车型数量（用于解决车辆删除后的车型的影响的判断） type\_num

CREATE TABLE t\_type\_info(

cartype VARCHAR(45) NOT NULL,

rentprice FLOAT NOT NULL,

type\_num INT NOT NULL,

PRIMARY KEY (cartype));

门店信息表 t\_store\_info

门店编号 storeid 门店名称 storename 门店地点 storepla

CREATE TABLE t\_store\_info (

storeid VARCHAR(45) NOT NULL,

storename VARCHAR(45) NOT NULL,

storepla VARCHAR(100) NOT NULL,

PRIMARY KEY (storeid));

管理员登录表 t\_store\_info

管理员id adminid 管理员密码 adminpsw

CREATE TABLE t\_admin (

adminid VARCHAR(45) NOT NULL,

adminpsw VARCHAR(45) NOT NULL,

PRIMARY KEY (adminid));

会员订单表： t\_user\_order

会员编号 uid 车辆编号 carid 租车时间 rentdate 还车时间 returndate 租车门店 rentstore 还车门店 returndate

支付金额 payment

CREATE TABLE t\_user\_order (

uid VARCHAR(45) NOT NULL,

carid VARCHAR(45) NOT NULL,

rentdate DATE NOT NULL,

returndate DATE NOT NULL,

rentstore VARCHAR(45) NOT NULL,

returnstore VARCHAR(45) NOT NULL,

payment FLOAT NOT NULL,

FOREIGN KEY (uid)

REFERENCES t\_user(uid)

on delete cascade

on update cascade,

FOREIGN KEY (carid)

REFERENCES t\_car\_info(carid)

on delete cascade

on update cascade,

FOREIGN KEY (rentstore)

REFERENCES t\_store\_info(storeid)

on delete cascade

on update cascade,

FOREIGN KEY (returnstore)

REFERENCES t\_store\_info(storeid)

on delete cascade

on update cascade);

车辆预约表： t\_car\_res

会员编号 uid 车型名称 cartype 租车时间 rentdate 还车时间 returndate 租车门店 rentstore 还车门店 returnstore

paydepo (押金统一为几百元)

CREATE TABLE t\_car\_res (

uid VARCHAR(45) NOT NULL,

cartype VARCHAR(45) NOT NULL,

rentdate DATE NOT NULL,

returndate DATE NOT NULL,

rentstore VARCHAR(45) NOT NULL,

returnstore VARCHAR(45) NOT NULL,

paydepo TINYINT NOT NULL,

FOREIGN KEY (uid)

REFERENCES t\_user(uid)

on delete cascade

on update cascade,

FOREIGN KEY (cartype)

REFERENCES t\_type\_info(cartype)

on delete cascade

on update cascade,

FOREIGN KEY (rentstore)

REFERENCES t\_store\_info(storeid)

on delete cascade

on update cascade,

FOREIGN KEY (returnstore)

REFERENCES t\_store\_info(storeid)

on delete cascade

on update cascade);

车辆租还表： t\_car\_rent

会员编号 uid 车辆编号 carid 租车时间rentdate

还车时间returndate租车门店rentstore还车门店returnstore

是否逾期（BOOL） overdue

CREATE TABLE t\_car\_rent (

uid VARCHAR(45) NOT NULL,

carid VARCHAR(45) NOT NULL,

rentdate DATE NOT NULL,

returndate DATE,

rentstore VARCHAR(45) NOT NULL,

returnstore VARCHAR(45),

overdue TINYINT,

FOREIGN KEY (uid)

REFERENCES t\_user(uid)

on delete cascade

on update cascade,

FOREIGN KEY (carid)

REFERENCES t\_car\_info(carid)

on delete cascade

on update cascade,

FOREIGN KEY (rentstore)

REFERENCES t\_store\_info(storeid)

on delete cascade

on update cascade,

FOREIGN KEY (returnstore)

REFERENCES t\_store\_info(storeid)

on delete cascade

on update cascade);

车辆所在门店：t\_car\_pla

车辆编号carid

车辆所在门店编号carstoreid

车辆是否出借（BOOL） carrent

CREATE TABLE t\_car\_pla (

carid VARCHAR(45) NOT NULL,

carstoreid VARCHAR(45) NOT NULL,

carrent TINYINT NOT NULL,

FOREIGN KEY (carid)

REFERENCES t\_car\_info(carid)

on delete cascade

on update cascade,

FOREIGN KEY (carstoreid)

REFERENCES t\_store\_info(storeid)

on delete cascade

on update cascade);

4、添加车辆前/后车型的变换

增加type\_num 每次车辆操作都会对车型产生影响

车型type\_num和车的总数的关系的约束

delimiter S$$

create procedure(in s1 varchar(45),in s2 varchar(45),in s3 varchar(45))

begin

insert into t\_car\_info values(s1,s2,s3);

update t\_type\_info set type\_num=type\_num+1 where t\_type\_info.cartype=t\_car\_info.cartype;

end$$

delimiter;

6、字段的元组约束 租还日期的大小关系

自增???????? 用？？？？？？？？

车辆编写函数自增 （ ）

删除车辆：

当删除车辆时，首先判断是否出借(car\_rent（t\_car\_pla）)，当租借则不能删除，没有租借则级联删除

1. 车型（触发器 车型数量减1 ） 当车辆数为0时不做处理
2. 车辆租赁表、用户订单表（无车辆编号的外键）

修改车辆“

首先判断是否出借(car\_rent（t\_car\_pla）)，当租借则不能更新，没有出借则级联更改??

1. 没有出借的画只能更改车牌

Over

添加车辆：

1. 车辆信息添加数据
2. 当有此车型 车辆数量加1，没有则添加车型，置数量为1

Admin21

private void button1\_Click(object sender, EventArgs e)

{

if(textBox1.Text != "" && textBox2.Text != "" && textBox3.Text != "")

{

Dao dao= new Dao();

string sql =$"insert into t\_car\_info values('{textBox1.Text}','{textBox2.Text}','{textBox3.Text}');update t\_type\_info set cartype=cartype+1 where t\_car\_info.cartype=t\_car\_info.cartype and t\_car\_info.carid='{textBox1.Text}'";

int n=dao.Execute(sql);

if (n > 0)

{

MessageBox.Show("添加成功");

}

else

{

MessageBox.Show("添加失败");

}

textBox1.Text = "";

textBox2.Text = "";

textBox3.Text = "";

}

加该类型数量

门店：

添加门店：

1. 门店信息添加数据

over

修改门店：

1. 可以修改名称以及地址

over

删除门店：

1、首先判断当前门店有无车辆

2、经上述判断car\_rent没有要删除的门店的id

3、car\_res表不做操作（rentstore和returnstore没有外键）

4、user\_order表不做操作（rentstore 和 returnstore没有外键）

5、门店信息表中添加属性（表示当前门店是否存在（为了方便用户进行订单查询））

select carid from t\_car\_rent,t\_store\_info where returnstore= '{id}' and t\_car\_rent.t\_store\_info=1 and returnstore=t\_store\_info.storeid;

门店的车辆信息：

use car\_rent;

select

c.carid,c.cartype,c.carlic,s.carrent

from

t\_car\_info c,t\_car\_pla s

where

c.carid = s.carid and s.carstoreid = '2003';

over

订单查询：

除了通过账单金额都能进行查询

select

u.rentdate,u.returndate,s1.storename,s2.storename

from

t\_user\_order u,t\_store\_info s1,t\_store\_info s2

where

s1.storeid = u.rentstore

and s1.storepla = '浙江省杭州市上城区金福桥路号';

over

预约车辆：

1、显示每辆车的所在门店（地址） 车辆信息（车牌不用） 车辆编号 车牌

已经出借不显示（！！！！！！！！！！！！！！！！）

2、直接插入t\_car\_res

取车：

1. 首先判断门店是否存在
2. 判断当前取车门店是否有该车辆

3、有车辆 （通过门店以及车型显示车辆的信息）

T\_car\_rent 插入数据

T\_car\_pla carrent

归还车辆：

1. 通过id 显示t\_car\_rent (carid,cartype,rentdate,rentstore 连接 t\_car\_res 显示预计的returndate,returnstore)
2. 进行还车：
   1. 将预约表删除（通过id）
   2. 将还车日期以及还车门店 是否逾期，确认租车金额变换的比例（逾期需要支付的金额变为1.2倍）
   3. 接受缴费信息
   4. 用户顶点表添加数据
   5. T\_car\_rent删除

返还支付金额

续约：

改预约表：

返还时间修改