

**数据库系统原理实践报告**

|  |  |
| --- | --- |
| 姓 名： | 余文梦 |
| 学 院： | 计算机学院 |
| 专 业： | 计算机科学与技术 |
| 班 级： | CS1408班 |
| 学 号： | U201414775 |
| 指导教师： | 瞿彬彬 |

|  |  |
| --- | --- |
| 分数 |  |
| 教师签名 |  |

2017年 月 日

**目录**

[1 课程任务概述 1](#_Toc644484069)

[2 实验一 基本SQL操作 2](#_Toc2052890979)

[2.1 任务要求 2](#_Toc994551173)

[2.2 完成过程 3](#_Toc287626857)

[2.2.1 数据定义 3](#_Toc461559377)

[2.2.2查询功能 4](#_Toc488928660)

[2.2.3选做部分 12](#_Toc1175922986)

[2.3任务总结 12](#_Toc1567012376)

[3 DBMS综合运用 13](#_Toc449043789)

[3.1 任务要求 13](#_Toc2144869401)

[3.2 完成过程 13](#_Toc493399384)

[3.3任务总结 13](#_Toc536762166)

[4 综合实践任务 14](#_Toc1227250420)

[4.1系统设计目标 14](#_Toc891125816)

[4.2 需求分析 14](#_Toc408317776)

[4.2.1 功能需求 14](#_Toc1333476982)

[4.2.2 性能需求 14](#_Toc548869058)

[4.3 总体设计 15](#_Toc2061816545)

[4.4 数据库设计 15](#_Toc1661459587)

[4.4.1 概念模型设计 15](#_Toc296796968)

[4.4.2 数据库的物理设计 17](#_Toc497018154)

[4.5 详细设计与实现 19](#_Toc1369706560)

[4.6 系统测试 19](#_Toc1191957385)

[4.7 系统设计与实现总结 19](#_Toc2110269052)

[5 课程总结 19](#_Toc400776610)

[附录 19](#_Toc1228237904)

# 1 课程任务概述

简要陈述介绍本实践课程的各项任务要求。

# 

# 2 实验一 基本SQL操作

## 2.1 任务要求

1）数据定义

参照下面的内容建立自己实验所需的关系数据

创建三个关系:

商品表【商品名称、商品类型】

GOODS【GNAME char(20),GTYPE char(10)】

主关键字为(商品名称)。商品类型为(电器、文具、服装等)

商场【商场名称,所在地区】

PLAZA【PNAME char(20)，PAREAchar(20)】

主关键字为商场名称。所在地区为(洪山、汉口、汉阳、武昌等)

销售价格表【商品名称、商场名称、当前销售价格、目前举办活动类型】

SALE【GNAME char(20),PNAME char(20),PRICE FLOAT,ATYPEchar(10)】

主关键字为(商品名称、商场名称)。举办活动类型为(送券、打折),也可为空值,表示当前未举办任何活动。表中记录如(‘哈森皮靴’,‘亚贸广场’,200,‘打折’),同一商场针对不同的商品可能采取不同的促销活动。

2）数据更新

（1）向上述表格中用 sql 语句完成增、删、个、改的操作;

（2）编写一个触发器,并测试该触发器;

（3）将 SALE 表中的打折记录插入到新表 SALE\_CHEAP 中,并基于 SALE\_CHEAP 表创建一个统计每个商场各自打折商品平均价格的视图。

3）用 SQL 语句完成下述查询需求:

（1）查询所有以“打折”方式销售的商品的当前销售情况,并按照价格的降序排列;

（2）查询所有没有任何活动的商品及其所在的商场,结果按照商品排序;

（3）查询价格在 200~500 元之间的商品名称、所在的商场名称、价格,结果按照商场名称排序;

（4）查询每种商品的最低价格、商品名称;

（5）查询以“送券”方式销售的商品总数超过 2 种的商场名称;

（6）查询以“送券”方式销售的商品总数超过 2 种的商场所在地区;

（7）查询价格为下列取值之一的商品名称、所在商场名称、目前举办活动的类型(88、188、288);

（8）查询以“老”字开头的所有商品的名称;

（9）查询同时销售“剃须刀”和“电池”的商场名称;

（10）查询不举办任何活动的商场;

（11）查询所销售的商品包含了“校园超市”所销售的所有商品的商场名称。

4）SQL 思考题部分(选做)

（1）SQLSERVER 的时间数据类型 datatime,可以记录形如“2010-5-1 13:30:15”的时刻,表示2010 年 5 月 1 日 13 点 30 分 15 秒时刻,通过查询 SQLSERVER 的帮助文件,解决下述 SQL语句的需求:

（1.1）在查询结果中将 datetime 属性的值转换成字符串类型;

（1.2）查询当前时刻之前 2 年、或者 2 个月、或者 2 天、或者 2 小时、或者 2 分钟的记录;

（1.3）查询某个时刻所在的年、月、日、时、分、秒;

（1.4）查询某个月的最后一天,返回 datetime 类型;

（1.5）假设在 SALE 表中有交易时间(DEELTM)属性,类型为 datetime,查出所有在整点时刻的交易,查出所有在整 15 分钟时刻(每个小时有 4 个整 15 分钟,例如 8:00、8:15、8:30、8:45)的交易。

（2）如何将查询结果的某些列的某些值转换成特殊的形式(例如成绩记录存储的都是百分制的成绩,对查询结果中 60 分以上的记录直接显示成绩,而对 60 分以下的记录则在成绩属性中显示“不及格”)

## 2.2 完成过程

### 2.2.1 数据定义

1. 创建一个MyGoods的数据库，然后使用create table命令创建三张表（商品表、商场表和销售价格表）。此部分代码如下：

/\*建表\*/

Create table GOODS(

Gname char(20) primary key,/\*商品名称\*/

Gtype char(10)/\*商品类别\*/

)COMMENT='商品表';

create table PLAZA(

Pname char(20) primary key,/\*商场名称\*/

Parea char(20)/\*商场所在区域\*/

)COMMENT='商场表';

create table SALE(

Gname char(20),

Pname char(20),

Price float,/\*价格\*/

Atype char(10),

primary key (Gname,Pname),/\*设定主码\*/

foreign key (Gname) references GOODS(Gname),

foreign key (Pname) references PLAZA(Pname)

)COMMENT='销售价格表';

（2）使用“insert into 表名 values(...);”语句完成数据的导入。导入完成后，GOODS表中的数据如图2. 1所示;PLAZA表中数据如图2. 3所示。接下来，使用这些数据完成指定的需求。



图2. 1 GOODS表

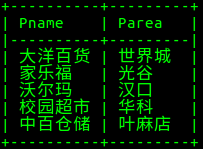


图2. 2 PLAZA表

### 2.2.2查询功能

（1）实验内容2.3部分。

功能需求：将 SALE 表中的打折记录插入到新表 SALE\_CHEAP 中,并基于 SALE\_CHEAP表创建一个统计每个商场各自打折商品平均价格的视图。

解决需求：首先，用create table语句创建一个SALE\_CHEAP表，该表的数据结构与表SALE一致。然后使用select语句将SALE表中打折的商品插入SALE\_CHEAP表中。实验代码如下所示。

/\*往打折表中插入数据\*/

INSERT INTO SALE\_CHEAP(GNAME,PNAME,PRICE,ATYPE)

SELECT GNAME,PNAME,PRICE,ATYPE

FROM SALE

WHERE ATYPE='打折';

执行结果生成的SALE\_CHEAP表如图2. 4所示。

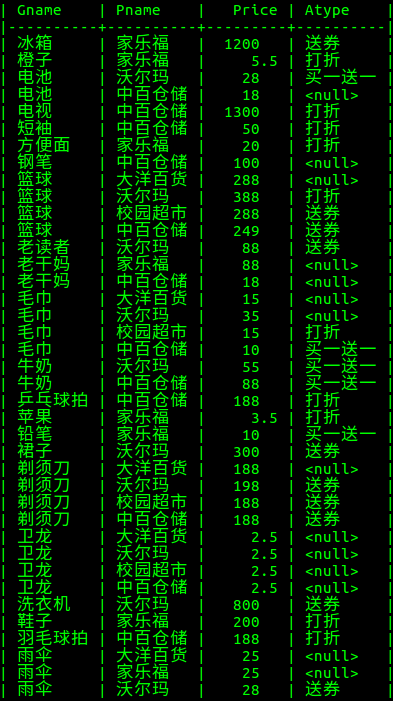


图2. 3 SALE表



图2. 4 SALE\_CHEAP表

接着，基于SALE\_CHEAP表，使用create view语句创建一个平均价格的视图，此部分实验代码如下，生成的视图结果见图2. 5

create view AVE\_PRICE /\*创建视图AVE\_PRICE\*/

AS SELECT Pname,AVG(Price) /\*计算平均价格\*/

from SALE\_CHEAP

group by Pname /\*按商场名字分组\*/



图2. 5 视图AVE\_PRICE

1. 实验内容3.1部分。

功能需求：查询所有以“打折”方式销售的商品的当前销售情况,并按照价格的降序排列;

解决需求：先从SALE表中查找出Atype为打折的商品，然后利用“order by”子句对查询结果进行降价排序。实验代码如下所示，执行结果见图2. 6。

select Gname,Price

from SALE

where Atype = '打折'

order by Price desc;/\*按价格降价排序\*/



图2. 6 查询结果1

1. 实验内容3.2部分。

功能需求：查询所有没有任何活动的商品及所在的商场,结果按照商品排序;

解决需求：此查询与上一个基本一致，新增了对IS NULL子句的使用。实验代码如下所示，执行结果见图2. 7。

select Gname,Pname

from SALE

where Atype IS NULL /\*判断Atype是否为空\*/

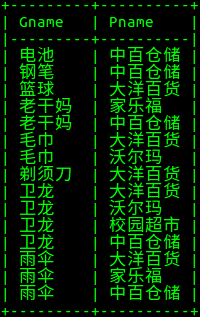


图2. 7 查询结果2

1. 实验内容3.3部分。

功能需求：查询价格在 200~500 元之间的商品名称、所在的商场名称、价格,结果按照商场名称排序;

解决需求：此查询新增了对比较运算符和逻辑与的使用。实验代码如下所示，执行结果见图2. 8。

select Gname,Pname,Price

from SALE where Price >= 200 and Price <= 500;



图2. 8 查询结果3

1. 实验内容3.4部分。

功能需求：查询每种商品的最低价格、商品名称;

解决需求：由于要求每种商品的最低价格，第一种思路是采用嵌套查询，第二种思路是采用group by子句先对表进行分组，然后调用MIN函数对每组进行求最小值。这里采用后一种思路，实验代码如下所示，执行结果见图2. 9。

select Gname,MIN(Price) /\*调用MIN子函数求最低价格\*/

from SALE

group by Gname;/\*按照名字分组\*/



图2. 9 查询结果4

1. 实验内容3.5部分。

功能需求：查询以“送券”方式销售的商品总数超过 2 种的商场名称;

解决需求：采用嵌套查询，先从PLAZA表中选出一个条目，然后在SALE表中找到对应商场的商品并且调用count子函数对活动类型为“送券”方式的商品进行计数，将计数结果大于2的输出。实验代码如下所示，执行结果见图2. 10。

select Pname

from PLAZA

where 2 < (

select count(\*) /\*调用count子函数进行计数\*/

from SALE

where SALE.Pname = PLAZA.Pname and Atype = '送券');



图2. 10 查询结果5

1. 实验内容3.6部分。

功能需求：查询以“送券”方式销售的商品总数超过 2 种的商场所在地区;

解决需求：基于上一个查询的实现，只需要将select所挑出的值改为Parea即可。实验代码基本一致，执行结果见图2. 11。



图2. 11 查询结果6



图2. 12 查询结果7

1. 实验内容3.7部分。

功能需求：查询价格为下列取值之一的商品名称、所在商场名称、目前举办活动的类型(88、188、288);

解决需求：此查询新增了对逻辑或（or）的使用。实验代码如下所示，执行结果见图2. 12。

select Gname,Pname,Atype

from SALE

where Price = '88' or Price = '188' or Price = '288';

1. 实验内容3.8部分。

功能需求：查询以“老”字开头的所有商品的名称;

解决需求：此查询新增了对正则规则的匹配，SQL语言中只支持任意字符“%”和任意单字符“\_”。实验代码如下所示，执行结果见图2. 13。

select Gname

from GOODS

where Gname like '老%';



图2. 13 查询结果8

1. 实验内容3.9部分。

功能需求：查询同时销售“剃须刀”和“电池”的商场名称;

解决需求：一开始，想到的是直接用INTERSECT子句对两个集合求交集找出同时满足两个条件的商品，却发现新版的MySql中不支持该子句，于是使用嵌套查询。实验代码如下所示，执行结果见图2. 14。

select Pname

from SALE

where Gname = '剃须刀' and

Pname IN(/\*在子查询中选出商品名称为电池\*/

select Pname

from SALE

where Gname = '电池'

);



图2. 14 查询结果9

1. 实验内容3.10部分。

功能需求：查询不举办任何活动的商场。

解决需求：先从SALE表中选出Pname和Atype两个属性列（用DISTINCT去重），并以此建立一个子表。接着利用嵌套查询找出有且仅有空的活动类型的数据项。实验代码如下所示，执行结果见图2. 15。

select Pname

from(/\*创建一个子表\*/

select DISTINCT Pname,Atype

from SALE

)as S1

where Atype NOT IN(

select Atype

from SALE

where SALE.Pname = S1.Pname and

SALE.Atype IS NOT NULL);

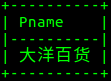


图2. 15 查询结果10

1. 实验内容3.11部分。

功能需求：查询所销售的商品包含了“校园超市”所销售的所有商品的商场。

解决需求：这是一个全称命题，而SQL语句中没有相应的语句，但是可以用存在量词对其进行表示。即改为“对于某商场，在校园超市里面，不存在某商品，该商场没有”。实验代码如下所示，执行结果见图2. 16。

select DISTINCT Pname

from PLAZA P1

where NOT EXISTS(

select \*

from SALE S1

where S1.Pname = '校园超市' and

NOT EXISTS(

select \*

from SALE S2

where P1.Pname = S2.Pname and

S2.Gname = S1.Gname)

);



图2. 16 查询结果11

经验证，上述查询结果全部与实验预期结果相符。

### 2.2.3选做部分

为了完成此部分，首先，在数据库中新建了一个SALE\_T表，其中在SALE表的基础上增加了时间属性。建表语句如下所示，插入一些基本数据后，得到的效果图如图2. 17所示。

create table SALE\_T(

Gname char(20),

Pname char(20),

Price float,

Atype char(10),

DEELTM DATETIME,

primary key (Gname,Pname))

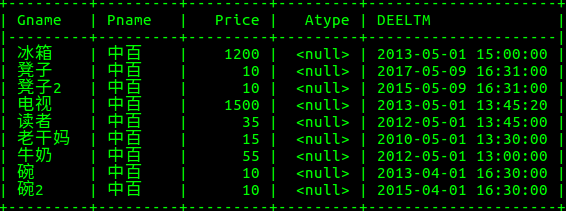


图2. 17 SALE\_T表内容

## 2.3任务总结

（总结实验中遇到的主要问题及所采用的解决方案）

# 

# 3 DBMS综合运用

## 3.1 任务要求

1. 学习sqlserver的两种完全备份方式：数据和日志文件的脱机备份、系统的备份功能（选做）。
2. 学习系统的身份、权限配置操作
3. 了解系统的查询性能分析功能（选做）

通过对第一部分任务中的SQL语句的查询性能分析，学习SQL查询的关系代数处理及其优化原理。

（4）了解SQLSERVER的数据导入导出功能（选做）

操作SQLSERVER的企业管理器集成界面，通过交互式方式实现数据的导入和导出。

（5）了解SQLSERVER的存储过程、触发器、函数实现过程

通过查看SQLSERVER的联机帮助文档和示例，体会存储过程、函数和触发器的原理，尝试编制具备基本功能的存储过程、触发器和函数的实际例子，查看其执行效果。

（6）了解SQLSERVER与常用应用程序开发环境集成的实现过程（选作）。

通过开发简单的DBMS应用程序，了解基本的数据库应用编程方法与技术，包括：数据库连接、嵌入式的数据读取操作、数据更新操作、事务提交技术等等。

## 3.2 完成过程

## 3.3任务总结

（注意：若有不止两项实验任务，则可继续增加章节，后续的综合实践任务五等等章节的编号顺延）

# 

# 4 综合实践任务

## 4.1系统设计目标

目前，虽然家庭拥有汽车量越来越多，但是当外出旅游或者办公时，租用汽车却是一个更好的选择。为了方便汽车租赁公司的数据管理，打算设计一个功能完善的汽车租借信息系统。完成用户、车辆、经手员工、租借情况、车辆损毁情况、交通违规罚款等信息的管理。

## 4.2 需求分析

### 4.2.1 功能需求

按照题目的要求，要实现的基本功能有以下几点：

1. 实现不同权限的浏览和更新。
2. 能够根据车辆使用情况计算押金退还金额。
3. 能查询客户的租借历史记录，并进行信誉度评价，进行会员制和非会员制的客户管理。
4. 能够管理车辆报修信息；
5. 能够生成租借公司的日、月、季度、年财务报表。

### 4.2.2 性能需求

为了达到上述要求，可以考虑此系统能有三种角色操作，即客户、普通员工和高级员工。他们拥有不同的权限，比如：客户可以查询自己的账户，租车记录等信息；普通员工可以进行用户信息和车辆信息中条目的增删改查等功能；高级员工可以进行所有信息的增删改查。在管理层面上，能够提供车辆信息界面，客户信息界面，租借服务界面以及员工信息界面等。

对于用户信息，能够查询用户编号、用户名、用户性别、用户年龄以及租车记录等信息。并且能够增对用户的历史记录进行评级。

对于员工信息，能够提供员工编号、员工姓名、性别、员工年龄以及员工处理的交易数和具体的交易编号等信息。并且能够对员工的工作历史进行等级评定。

对于车辆信息，能够查询到车辆的车辆编号、车型、颜色、损毁程度、目前状态（空闲或者在使用）和以往租借记录等信息。

对于租借信息，可以查询所有交易信息，包括交易编号、交易经手员工、用户编号、起止时间等信息。

另外，有一个对外的公共查询窗口，可以方便大众查看汽车的基本信息，而不用登录系统，使得该系统更加合理化、人性化。

## 4.3 总体设计

阐述系统的总体设计方案，包括系统的C/S或者B/S架构图、功能模块组成及其说明、系统的总体业务流程图及其说明等等总体设计的主要内容。

## 4.4 数据库设计

### 4.4.1 概念模型设计

首先设计各个分E-R图，其中，用户信息的E-R图如图3. 1所示；员工信息E-R图如图3. 2所示；车辆信息的E-R图如图3. 3所示；租借信息的E-R如图3. 4所示。特别地，对于主码用了下划线进行标识。

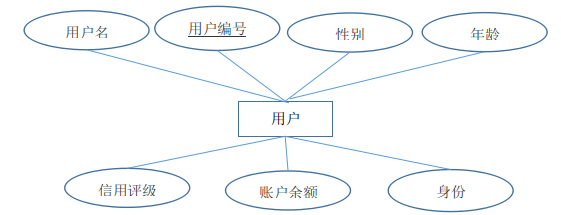


图3. 1 用户信息E-R图

备注：信用评级

A——优——免交押金

B——良好——交正常押金

C——较差——交1.2倍押金

D——差——交双倍押金

身份(账户余额为1000及以上可享受vip服务）

0——普通客户——按照正常价格计费。

1——会员——享八折优惠，并且拥有会员专享特权，即可以使用会员专享车辆

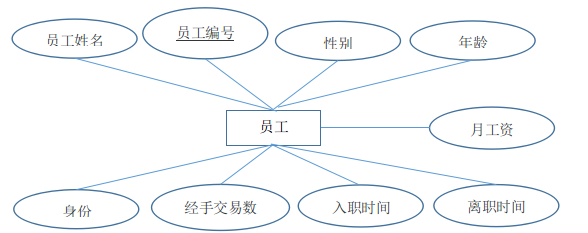


图3. 2 员工信息E-R图

备注：身份

0——普通员工

1——高级员工

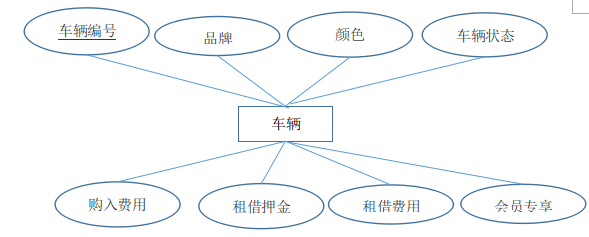


图3. 3 车辆信息E-R图

备注：车辆状态

A——空闲

B——正在使用

C——维修中

会员专享

Y——为会员专享车辆

N——非会员专享车辆

租借费用为每半个小时的价格，不足半个小时按照半个小时计算。

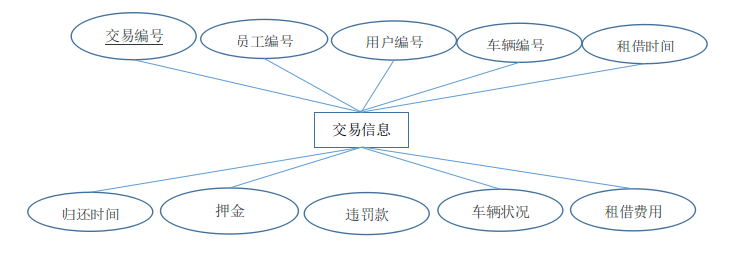


图3. 4 租借信息的E-R图

备注：车辆状况

A——优——正常缴费，并且提升用户信用评级

B——良——正常费用的1.2倍

C——一般——正常费用的1.5倍

D——差——正常费用的2倍，并且降低用户信用评级

然后，进行总体E-R图设计，如图3. 5所示。

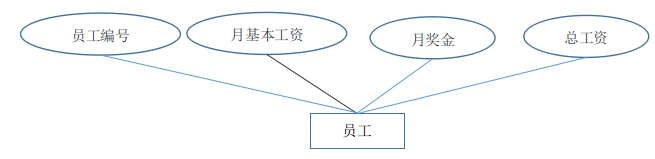


图3. 5 员工工资的E-R图

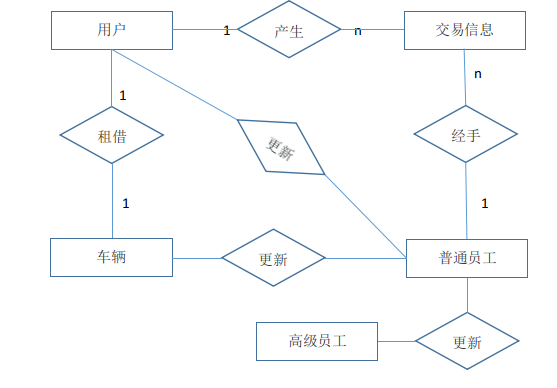


图3. 6 系统总体E-R图

### 4.4.2 数据库的物理设计

完成了上述设计之后，便可以开始数据库的物理设计。考虑到程序设计的简易性和通用性，本系统采用MySql提供后台数据库支持，在其中创建了下列表。

表3. 1 用户信息表

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 长度与格式 |
| 用户名（Uname） | VarChar | 20 |
| 用户密码（Ucode） | VarChar | 12 |
| 性别（Sex） | Char | 1（M/W） |
| 年龄（Age） | Int | 短整数 |
| 信用评级（Cre） | Char | 1 |
| 账户余额（Acn） | Float | 浮点数 |
| 身份（Ide） | Char | 1（0/1） |
| 主码：用户编号（Unum） | | |
| 外码：无 | | |

表3. 2 员工信息表

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 长度与格式 |
| 员工姓名（Wname） | VarChar | 20 |
| 员工编号（Wnum） | VarChar | 12 |
| 员工密码（Ucode） | VarChar | 12 |
| 性别（Sex） | Char | 1（M/W） |
| 年龄（Age） | Int | 短整数 |
| 交易数（Tnum） | Int | 短整数 |
| 入职时间（Ctime) | DateTime | 时间 |
| 离职时间（Ftime） | DateTime | 时间 |
| 身份（Ide） | Char | 1（0/1） |
| 主码：员工编号（Wnum） | | |
| 外码：无 | | |

表3. 3 员工工资表

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 长度与格式 |
| 员工编号（Wnum） | VarChar | 12 |
| 月基本工资（Bsal） | Float | 浮点数 |
| 月奖金（Pride） | Float | 浮点数 |
| 总工资（Sal） | Float | 浮点数 |
| 主码：员工编号（Wnum） | | |
| 外码：员工编号（Wnum） | | |

表3. 4 车辆信息表

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 长度与格式 |
| 车辆编号（Cnum） | VarChar | 12 |
| 品牌（Brand） | VarChar | 12 |
| 颜色（Color） | VarChar | 12 |
| 车辆状态（Sta） | Char | 1（A/B/C） |
| 购入费用（Fee） | Float | 浮点数 |
| 购买时间（Btime) | DateTime | 时间 |
| 押金（Cash） | Float | 浮点数 |
| 租借费用/h（Ree） | Float | 浮点数 |
| 会员专享（Vip） | Char | 1（Y/N） |
| 主码：车辆编号（Cnum） | | |
| 外码：无 | | |

阐述数据库设计的主要内容，依次是ER图设计及其说明、数据库逻辑结构设计（包括表、视图的逻辑结构说明，含主码、外码说明、用户定义完整性）、数据库物理设计（例如索引）。

## 4.5 详细设计与实现

阐述各主干功能的实现过程，包括主干功能的业务流程图、关键技术和算法说明、数据库事务的定义与实现、数据库函数和触发器的定义与实现等（不允许大段引用源码，如有必要引用必须加详细注释）。

## 4.6 系统测试

包括对测试数据的说明、测试过程阐述、测试结果分析。

## 4.7 系统设计与实现总结

分条目概括、总结应用系统设计的主要工作。

# 5 课程总结

逐条概括、总结此次课程实践的主要工作，阐述此次课程实践的心得体会，展望此次课程实践的有待改进和完善的工作。

# 附录