

校园二手交易系统的数据库分析与设计

潘星宇 曾 丽

成都锦城学院计算机与软件学院 四川 成都 611731

【摘要】随着计算机技术的不断发展,计算机技术已经深入到社会生活的各个方面,校园二手交易系统能够快速进行商品搜索以及线上交易等操作,简化校园二手交易的过程,更大提高效率。本文基于数据库系统的基础,对系统进行需求分析、数据库设计、数据库实现、数据库测试和系统维护等方面的阐述,完成对校园二手交易系统的数据库建立。

【关键词】二手交易系统;数据库;DBMS

1 引言

随着校园二手交易的需求不断提升,同时目前校园的二手交易只能通过寻找卖家,再通过联系卖家了解商品信息过后再进行交易,这种方法不仅效率低、过程麻烦,并且不容易找到适合自己的商品。为了校园二手交易更加方便,满足学生能够更加容易找到自己满意的商品或者卖出自己的商品,此校园二手交易系统能够提高交易的效率,简化交易过程。校园二手交易系统可以建立更大的数据库用于保存买家、卖家以及商品的所有信息,并且提高了搜索、查询信息的速度,方便了学生二手交易信息交流的过程。

2 需求分析

本校园二手交易系统是针对学生在校进行二手交易的管理,面向的用户主要分为卖家和买家。

买家在进入系统时首先进行买家注册,系统能够保存自己的买家信息,同时对于自己想要购买的商品能够在商品表中进行查询、在购买商品过后根据消费的总金额可以进行更新会员信息,根据不同的会员等级能够享受不同的折扣,对于买家的信息能够随时进行修改,买

家通过下单能够更新订单表的信息,同时根据订单表中的信息在商品表中进行订购,在完成交易后买家能够对商家进行评价。卖家进入系统时进行卖家注册,能够保存自己的卖家信息,对于自己想要卖出的商品进行商品注册,将商品信息增加到商品表中,在完成交易过后,商品表中的交易商品信息立即更新,并且能够同时修改商家的信息,商家能够修改自身的部分信息。

3 系统设计的相关技术

Microsoft SQL Serve 数据库是一款具有使用方便可伸缩性好与相关软件集成程度高等优点的全面数据库平台,SQL 语言是一种数据库查询和程序设计语言,能够用于存取数据以及查询、更新和管理关系数据库系统,通过 SQL 语言能够完成对数据库中数据的一系列操作。

4 数据库设计

4.1 系统 E-R 图设计

通过需求分析,本系统包含有买家、卖家、商品、订单实

体,根据各实体之间的关系描述设计出系统的 E-R 图如图 1:

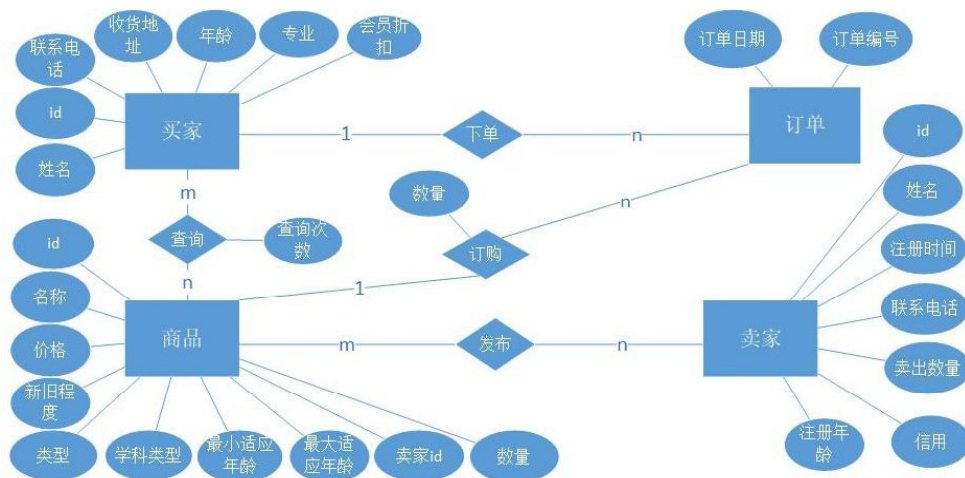


图 1 二手交易系统 E-R 图

4.2 逻辑结构设计

通过 E-R 模型向关系模型进行转换,转换后的关系

模式为:

卖家 (id, 姓名, 注册时间, 注册年龄, 联系电话,

信用, 卖出数量)

买家 (id, 姓名, 年龄, 会员折扣, 联系电话, 收货地址, 专业)

商品 (id, 卖家 id, 名称, 价格, 新旧程度, 最小适应年龄, 最大适应年龄, 类型, 学科类型, 数量)

订单 (订单编号, 订单日期, 买家 id, 商品 id, 数量)

买家浏览商品表 (买家 id, 商品 id, 查询次数)

根据转换为的关系模式, 在 Microsoft SQL Serve 中建立数据库, 数据库表包含卖家表、买家表、商品表、订单表、买家浏览商品表。其中买家表如表 1 所示, 订单表如表 2 所示。

表 1 买家表

字段名	字段类型	是否空值	备注
id	int	N	主键
姓名	char(8)	N	
年龄	int	N	
会员折扣	float	N	
联系电话	char(11)	N	
收货地址	varchar(20)	N	
专业	varchar(20)	N	

表 2 订单表

字段名	字段类型	是否空值	备注
订单编号	int	N	主键
订单日期	date	N	
买家 id	int	N	外键
商品 id	int	N	外键
数量	int	N	

4.3 物理设计

根据该系统的关系模型从而实现数据库的建立, 需要对该系统进行物理文件的安排, 同时建立相关的索引, 基于各表中的主键进行查询操作。

4.4 数据安全性设计

数据的安全性用于保证数据库的安全性和数据完整性, 防止数据库中的数据丢失、数据被恶意窃取或修改等破坏数据库以及影响用户正常使用行为, 能够有效保护所有的数据不会被破坏或丢失等。本系统会根据不同的用户设置不同的权限, 对不同的数据库数据设置不同的权限, 对访问数据库的用户进行身份验证, 对于重要、私密的数据进行加密保护, 并且对数据进行多份保存防止数据的丢失。

5 数据库实现

5.1 数据库表

根据需求分析, 首先分别建立买家、卖家、商品、订单四个实体的信息表, 同时再根据买家查询商品产生的买家商品浏览表实体建立数据库表格, 部分代码如下:

```
create table 卖家
```

```
(id int primary key check (id between 1 and 99999999) identity(1,1),
```

```
-- 将 id 设置为主键, 并且设置自动递增
```

```
姓名 char(8) not null,
```

```
注册时间 date,
```

```
注册年龄 int check(注册年龄 between 0 and 100) default 0,
```

```
联系电话 char(11) not null,
```

```
信用 int check(信用 between 1 and 100) default 100 not null,
```

```
卖出数量 int check (卖出数量 between 0 and 99999) default 0 not null)
```

5.2 存储过程

根据建立数据库的存储过程可以快速实现对数据的各种操作, 能够提高本系统各种对数据操作的效率。通过建立存储过程, 买家可以快速查询卖家的信息以及商品的信息, 并且根据查询记录买家的查询次数到买家浏览商品表, 在注册买家信息时, 可直接插入自身信息到买家表中, 卖家能够向卖家表插入自己的卖家信息以及向商品表插入新增的商品信息。在进行交易时, 买家下单商品, 完成对订单的更新, 依据买家的 id, 以及商品的 id, 将订单信息插入订单表中。根据订单信息, 进行商品订购, 在完成交易后买家可以对卖家进行评价, 同时更新卖家的信用等信息。系统可以根据买家的年龄、专业、浏览商品的记录给买家推送商品信息。建立存储过程后, 可以完成以上等功能实现, 部分代码如下:

```
create proc 卖家注册 -- 分别输入卖家姓名、卖家联系电话, 注册时间使用 getdate 生成, id 自动递增生成, 其余属性为默认值
```

```
@x char(8), @y char(11)
```

```
as
```

```
insert into 卖家 (姓名, 联系电话, 注册时间) values (@x, @y, getdate())
```

```
create proc 订单评价
```

```
@a int, @b int, @c int -- @a 代表买家 id, @b 代表商品 id, @c 代表评分 (0-100)
```

```
as
```

```
declare @d int -- @d 代表卖出数量
```

```
select @d = 购买数量 from 订单 where 买家 id = @a and 商品 id = @b
```

```
update 卖家 set 卖家.信用 = (卖家.信用 + @b) / 2 where 卖家.id = (select 卖家 id from 商品 where id = @b)
```

```
update 卖家 set 卖家.卖出数量 = 卖家.卖出数量 + @d where 卖家.id = (select 卖家 id from 商品 where id = @b)
```

```
create proc 订单金额
```

```
@a int, @b int -- @a 代表买家 id, @b 代表商品 id
```

```
as
```

```
select 订单.购买数量 * 商品.价格 from 订单 inner join 商品 on 订单.商品 id = 商品.id where 买家 id = @a and 商品 id = @b
```

5.3 触发器

通过建立触发器, 在对系统中对象进行插入、删除、修改操作前后, 可以完成指定操作, 本系统建立后触发器, 针对在完成订单过后, 对商品表中该商品数量进行修改, 同时根据订单中买家购买总金额更新买家的会员折扣信息, 代码如下:

```

create trigger t1
on 订单
after insert
as
declare @a int,@b int
select @a= 商品 id from inserted
select @b= 购买数量 from inserted
update 商品 set 数量 = 数量 -@b where id=@a
create trigger t2
on 订单
after insert
as
declare @a int,@b int
select @a= 买家 id from inserted
select @b=sum( 订单. 购买数量 * 商品. 价格 )
from 订单 inner join 商品 on 订单. 商品 id=商品. id
where 订单. 买家 id=@a
update 买家 set 会员折扣 =(
case
when @b between 500 and 1000 then 0.9
when @b between 1000 and 3000 then 0.85
when @b between 3000 and 10000 then 0.8
when @b>10000 then 0.75
end)where id=@a;

```

6 数据库测试

6.1 输入数据测试

增加新的买家信息, 运行检测新增买家信息功能:

```
insert into 买家 (姓名, 年龄, 联系电话, 收货地址, 专业) values('阿威', 18, '12345678916', 'A', '计算机系')
```

增加新的卖家信息, 运行检测新增卖家信息功能:

```
insert into 卖家 (姓名, 联系电话) values('威威', '12345678922')
```

增加新的商品信息, 运行检测新增商品信息功能:

```
insert into 商品 (卖家 id, 名称, 价格, 新旧程度, 最小适应年龄, 最大适应年龄, 类型, 学科类型) values(6, '手机', 800, 85, 18, 30, '电子产品', '无')
```

6.2 完成相关操作测试:

如果一位 id 为 1 的买家想要购买一个平板电脑, 那么他的购买操作如下:

在商品表中查询平板电脑的信息: `select *from 商品 where 名称='平板电脑'`

在选中商品类型过后, 下单生成订单信息:

```
insert into 订单 (买家 id, 商品 id, 数量, 订单日期) values(1, 11, 1, getdate())
```

通过订单在商品表中订购, 将该商品信息修改:

```
update 商品 set 数量 = 数量 -1 where id=11
```

商家通过订单中买家 id, 得到买家表的该买家的联系电话以及收货地址:

```
select 联系电话, 收货地址 from 买家 where id=1
```

7 系统实施与维护

对于本系统的实施需要建立数据库的安全性, 来保障用户数据的安全, 并且能够保证用户的查询信息的快捷性和全面性。需要不断完善数据库各方面的维护功能, 从而建立强大快捷的搜索功能, 对一定的数据进行更多的分类, 并且增加更多的检索功能, 从而加快数据的查询速度。通过建立符合各种不同场景的视图, 来保障用户信息的安全。对于系统的维护, 研究设计的数据库系统在进行运行后, 会随时间推进还会产生很多漏洞, 并且在系统运用过程中, 会随着用户的需求不断产生更多的功能需求, 就需要不断的对该数据库系统不断进行更新维护, 解决系统出现的各种漏洞以及功能需求。

对于系统的维护, 主要对本数据库进行校正性的维护、适应性的维护、完善性的维护以及预防性的维护这些方面维护。定期地对系统进行校正性维护以确保系统的各项功能能够完整运行、并且保证系统不出现漏洞。对于系统的环境进行系统的适应性维护, 能够帮助系统能够适应环境保证功能的完整运行。根据用户的需求不断更新, 在原有功能的基础上进行升级并且新增更多的功能, 保证满足用户的各项需求以及效率的提高。为了保证系统在未来的可维护性和可行性的、保证系统功能能够持续运行进行预防性的维护。

8 总结

本文基于数据库的基础功能完成校园二手交易系统数据库设计, 通过建立对象表、建立存储过程以及建立触发器等操作, 根据需求分析、数据库设计、数据库实现、数据库测试和系统维护等方面完成对本系统的设计和实现。本系统能够满足对校园内学生的二手交易需求, 极大提高了效率, 能够有效地组织、存储和管理数据。其系统数据库的实际应用效果需在不断验证后, 并且获取用户的需求和感受, 不断改进功能, 针对系统的不足不断加强优化。

【参考文献】

[1] 赵亮, 胡树煜, 吴宇玲. 校园旧物交易系统的财务数据库设计 [J]. 中国管理信息化, 2021, 24(7): 63-64

[2] 张振, 王朝阳. 大学闲置物品交易平台后端的设计与实现 [J]. 内蒙古科技与经济, 2021, 1: 91-92