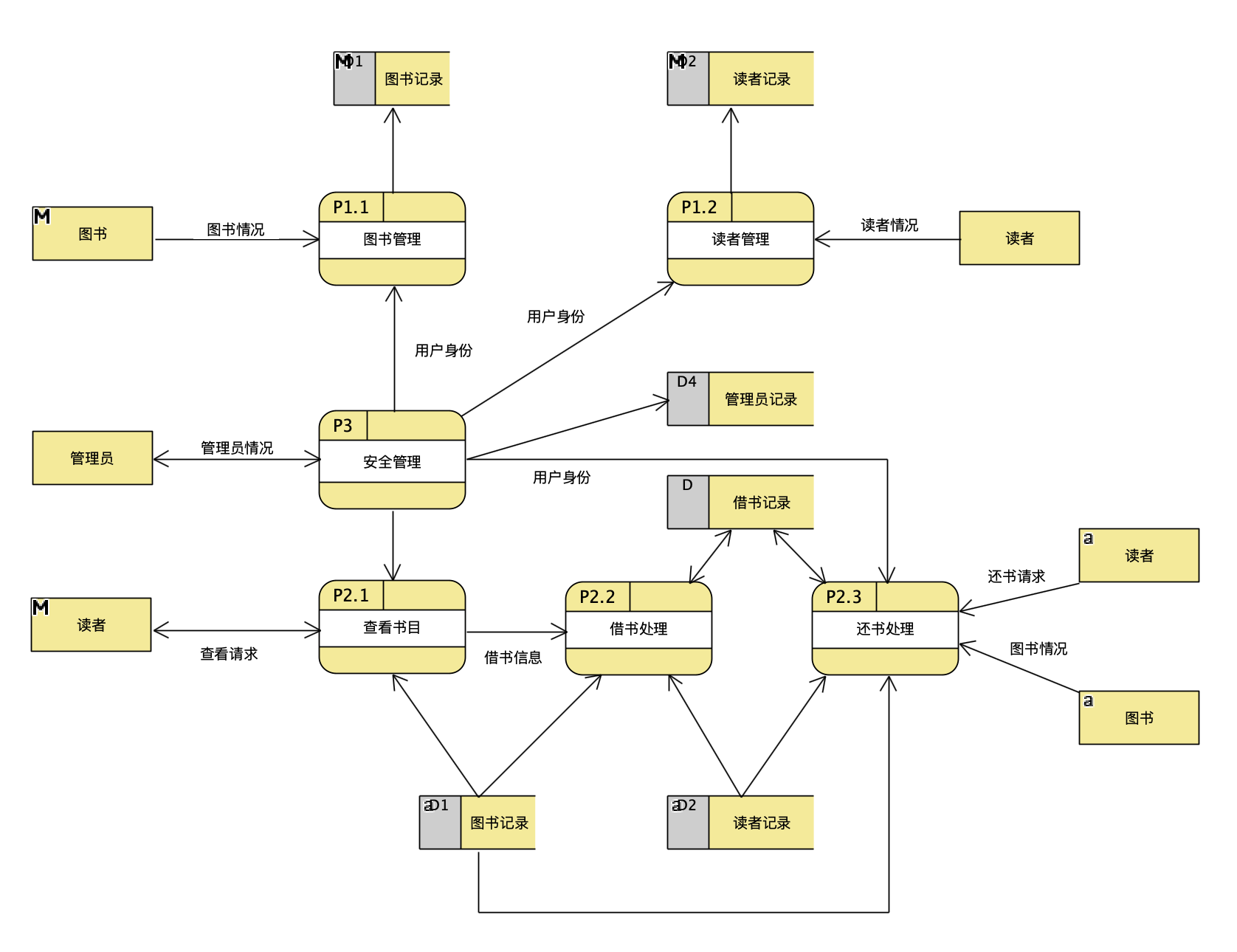
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 《数据库原理》实验报告 | | | | |
| 题目：实验八  数据库和表的创建与管理 | 学号 | 姓名 | 班级 | 日期 |
| **201730 0138** | **严愉程** | **HC001706** | **2019.11.21** |

1. 实验内容、步骤以及结果

1.需求分析

使用visual Paradigm绘制数据流图



数据流：

(1)图书情况数据流

数据流名：读者情况

说 明：新书入库，建立图书记录；

还书时检查书籍的基本信息以及是否完好。

来源去向：图书→P1.1, 图书→P2.3

数据结构:图书信息表

(2)读者情况数据流

数据流名:读者情况

说 明:根据读者情况建立读者记录;

来源去向:读者→P1.2

数据结构: 用户名+口令，读者信息表

(3)管理员情况数据流

数据流名:管理员情况

说 明:通过用户名和口名鉴别管理员身份。

来源去向:管理员→P3

数据结构:用户名+口令，管理员信息表

(4)用户身份数据流

数据流名:用户身份

说 明:不同的用户身份进入的处理过程不同。

来源去向:P3→{P1.1, P1.2, P2.1, P2.3}

数据结构:用户信息表(非法用户/读者/管理员)

(5)查看请求数据流

数据流名:查看请求

说 明:通过书名和类别查询库中的图书。

来源去向:读者→P2.1

数据结构:类别/书名，图书信息表

(6)借书信息数据流

数据流名:借书信息

说 明:根据书号和读者编号，进行借书处理。

来源去向:P2.1→P2.2

数据结构:书号 + 读者编号

(7)还书请求数据流

数据流名:还书请求

说 明:确认图书正确和完好后，删除借书记录。

来源去向:读者→P2.3

数据结构:图书信息表，读者信息表

数据字典

(1)D1: 图书记录

――――――――――――――――――――――――――――――――数据存储名:图书记录

说 明:记录每一本图书的信息，书号是码。

输 入: P1.1

输 出: P2.1, P2.2, P2.3

数据描述:书号,类别,出版社，作者，署名，定价，借出与否

数据 量:数据量约250000条

存取频度:平均每天1000次。

存取方式:联机处理;检索和更新;主要是随机检索。

(2)D2: 读者记录

――――――――――――――――――――――――――――――――数据存储名:读者记录

说 明:记录每一个读者的信息，编号是码，性别只能是“男”或“女”。

输 入:P1.2

输 出:P2.2,P2.3

数据描述:编号,姓名，单位，性别，电话

数据 量:数据量约15000条

存取频度:平均每天500次。

存取方式:联机处理;主要是检索处理，以随机检索为主。

(3)D3: 借书记录

――――――――――――――――――――――――――――――――数据存储名:借书记录

说 明:记录借书还书的信息，读者编号和书号为外码。

输 入:P2.2

输 出:P2.2,P2.3

数据 描述:读者编号,书号，借阅日期

数据 量:数据量约50000条

存取 频度:平均每天1000次。

存取 方式:联机处理;主要是更新处理;随机检索。

(4)D4: 用户记录

―――――――――――――――――――――――――――――――

数据存储名:用户记录

说 明:记录用户的信息，主要是区分级别。

输 入:P3

输 出:P3

数据 描述:用户名,密码，级别

数据 量:数据量约1000条

存取 频度:平均每天100次。

存取 方式:联机处理;主要是检索处理;顺序检索。

主要处理过程

(1)P1.1: 图书管理处理过程

处理过程名:图书管理

说 明:新入入库时，对新书进行编码，并且存储信息。

输 入:图书情况，用户身份

输 出:D1

(2)P1.2: 读者管理处理过程

处理过程名:读者管理

说 明:建立读者信息表，对读者编号。对读者信息进行维护。

输 入:读者情况，用户身份

输 出:D2

(3)P2.1: 查看书目处理过程

处理过程名:查看书目

说 明:根据输入信息对图书的信息进行查询。

输 入:查看请求，D1,用户身份

输 出:查看请求，借书信息

(4)P2.2: 借书处理过程

处理过程名:借书处理

说 明:确认读者符合借书条件，办理借书手续。

输 入:图书信息，D1,D2,D3

输 出:借书信息

(5)P2.3: 还书处理过程

处理过程名:还书处理

说 明:对照对应的借书卡，确认书号和书无误时办理还书手续。

输 入:D1,D2,D3，还书请求，图书情况，用户身份

输 出:D3

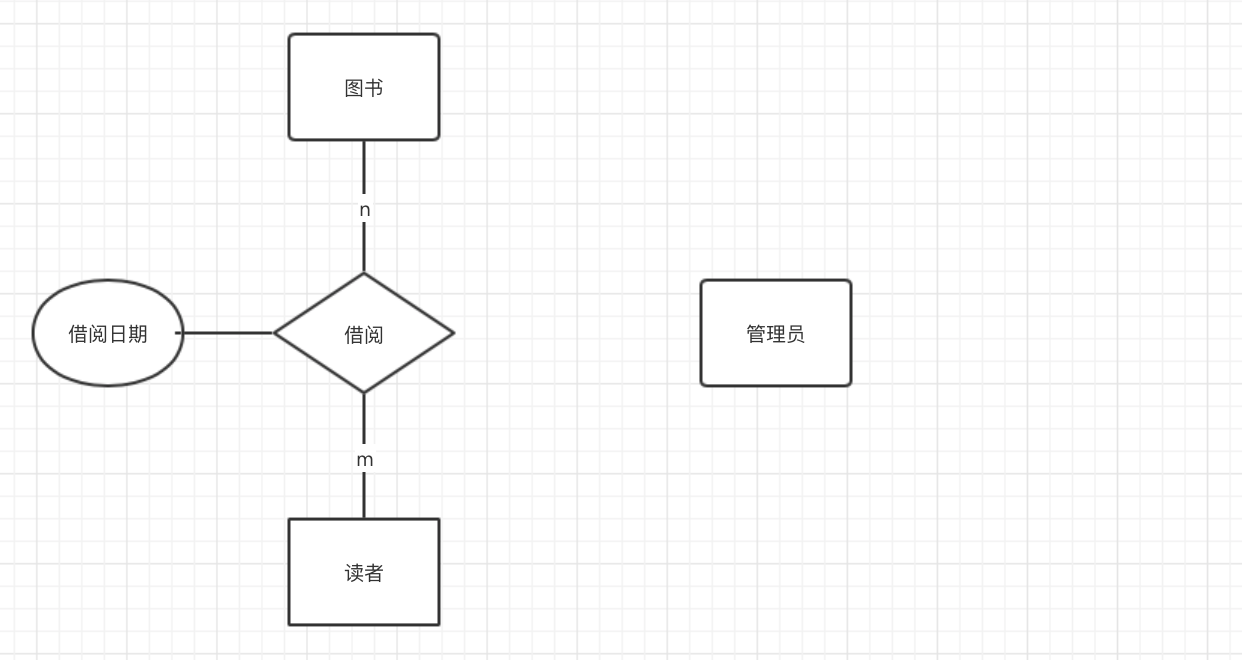
(6)P3: 安全管理处理过程

处理过程名:安全管理

说 明:通过用户名和口令，确认用户身份。

输 入:管理员情况，D4

输 出:D4,用户身份，管理员情况

2.概念结构设计

为了ER图的简洁性，这里省略了部分属性

3.逻辑结构设计

1) E-R图向逻辑模型转换

数据库名：图书馆管理系统

读者：（读者编号，）

图书：（图书编号，）

管理员：（管理员编号，）

借阅：（读者编号，图书编号，借阅时间，应还时间，归还时间，是否过期）外码：读者编号，图书编号

2）关系模式的优化 - 函数依赖集

Fadmin = {adminId→adminName，adminId→adminPasswd, ……}

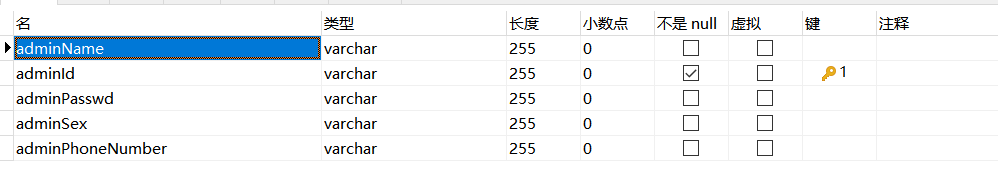
Fbooks = {bookId→bookName，bookId→author，……}

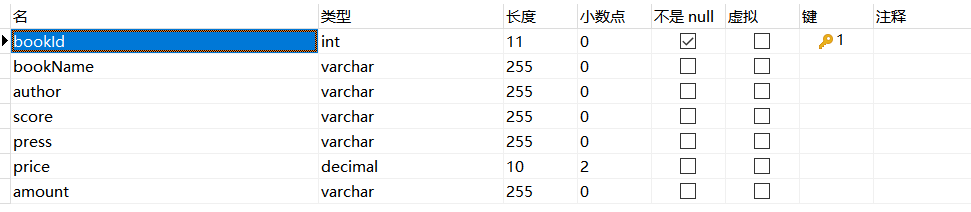
Fborrowlist = {borroNumber,userId,bookId→bookName，borroNumber,userId,bookId→Name}

Fusers = {userId→userName，userId→userPasswd，……}

不存在非主属性和主属性对码的部分函数依赖 和传递函数依赖，因此满足BCNF。

4.数据库的物理设计









其中borrowlist的外键依赖于users的userId和books的bookId。对于update是级联操作，对于delete是restrict

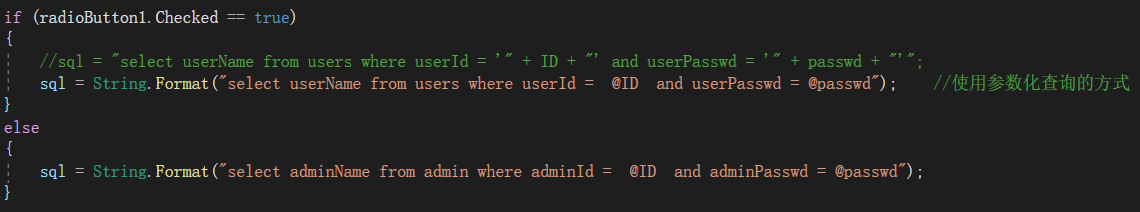
五、数据库的实施

1.项目简介

一个简单的图书管理系统包括图书馆内书籍的信息、学校在校学生的信息以及学生的借阅信息。此系 统功能分为面向学生和面向管理员两部分，我划分成了两个种类型的账号，分别放在表users和表 admin中。（注：本项目中的学生等价于用户，教师等价于管理员）

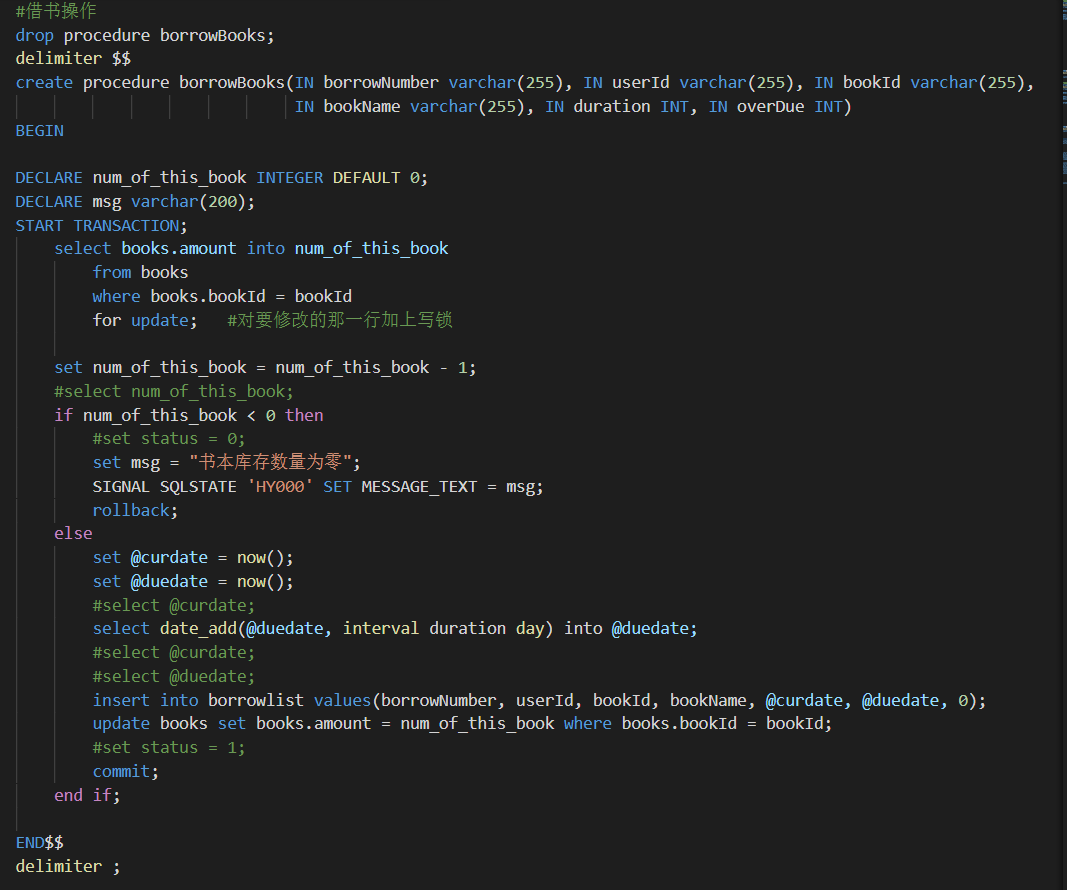
为了防止sql注入，在本项目中我采用了参数化查询的方式。

登录部分使用的关键语句为：



2.用户部分

书籍借阅：因为可能存在多个学生一起借同一本书的情况，所以我使用存储过程+事务的方式来进行 处理，并且为了防止并发过程中不出错，我还在修改书籍数量的地方加上了行级锁的处理。



书籍查询: 使用select语句设计了正常的书籍查询操作

书籍归还: 普通的书籍归还操作

书籍续借: 用户可以对自己已借的书籍进行续借

3.管理员部分

学生增加：添加一个用户到users表中

学生删除：检查是否还有书籍没有归还，如果有书籍没有归还则不同意被删除。如果图书都已经归还则从user表中删除。

学生信息修改：修改user表中的信息。

书籍入库：添加书籍到books表中

书籍删除：检查是否还有书籍还没归还，只有当全部归还了才能删除书籍

书籍信息修改：修改books表中的书籍信息

对学生借阅、续借、归还的确认：我觉得这个功能实际中不太现实就没有加

书籍查询：对库中的书籍进行查询

用户查询：对库中的用户进行查询

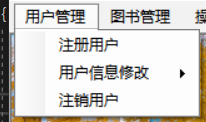
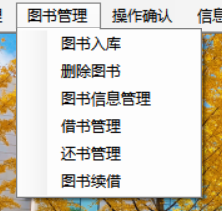
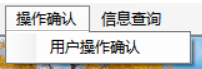
4.GUI展示

登录界面



主界面



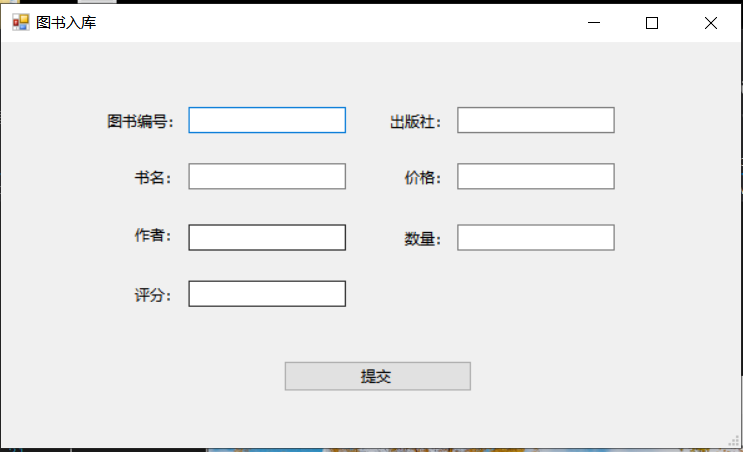
各个菜单栏的操作



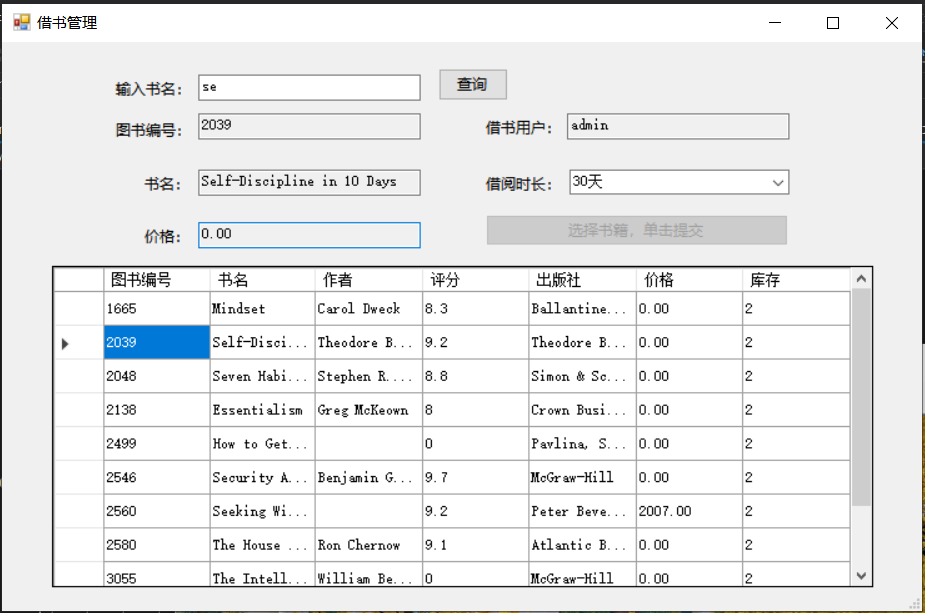
如果使用用户账号登录，菜单栏里的部分功能会禁止使用：

主要功能GUI界面

图书入库



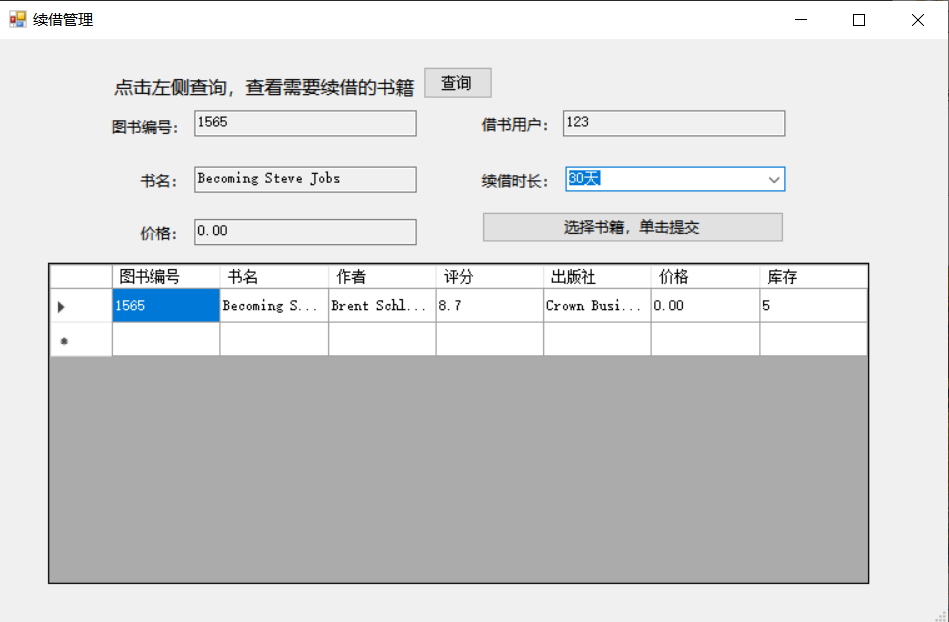
图书借阅



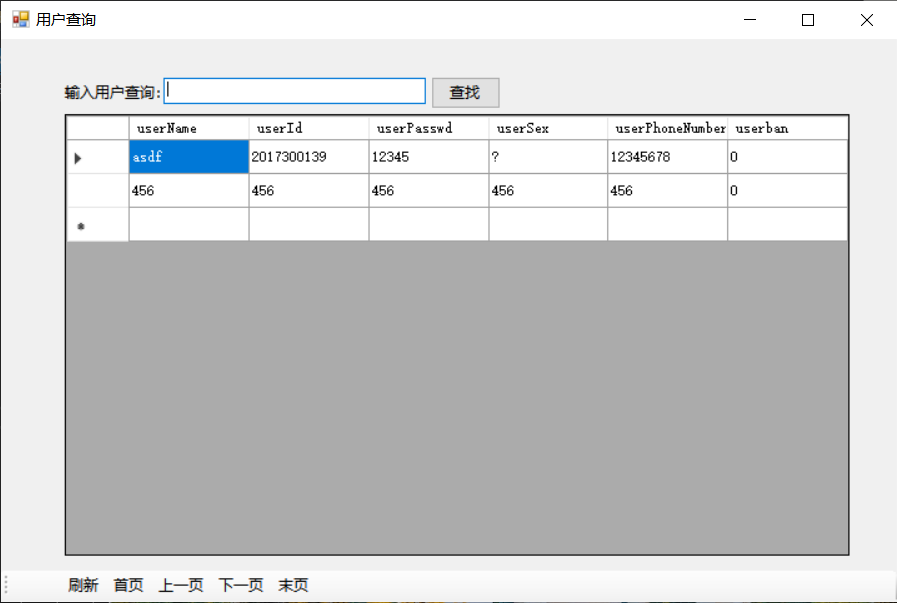
图书归还



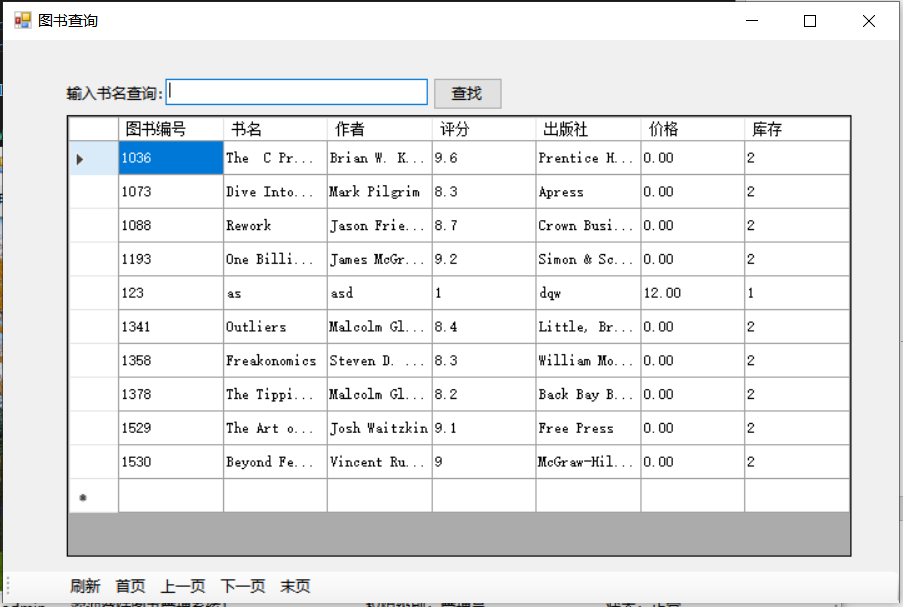
图书续借



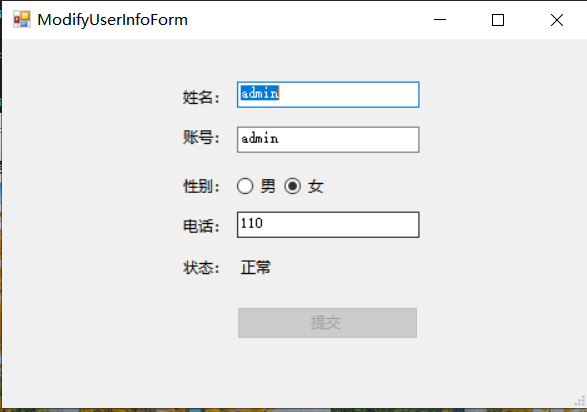
用户查询



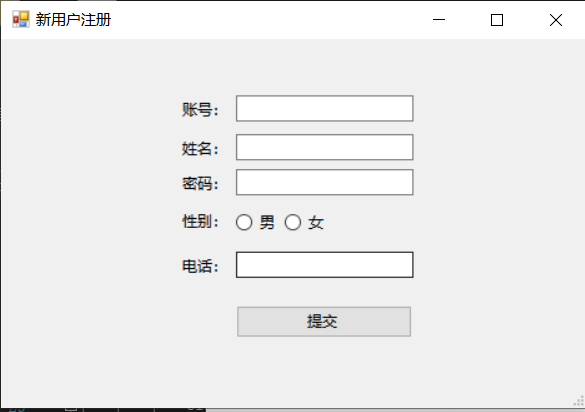
图书查询



用户信息修改



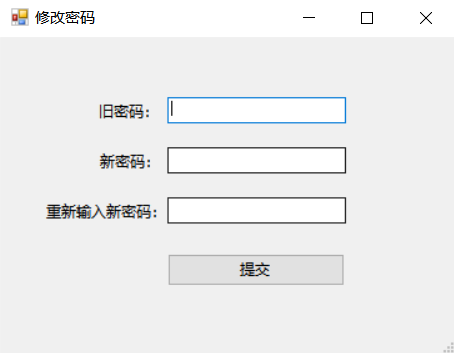
注册用户



注销用户



修改密码



5.具体操作方法

详见演示视频

1. 实验中出现的问题以及解决方案(对于未解决问题请将问题列出来)

除了标题内容以外，该部分内容中还可以写对于实验的一些感受，建议，意见等。

**学到了很多。学习了数据的应用系统的开放流程：从需求分析到概念结构设计，从逻辑设计到物理设计，最后编写代码进行数据库的实施和维护。**

**学习了C#编程。学习了通过拖控件的方式进行GUI界面的设计。学习了ODBC的开发。**

**将课堂上所学的数据库只是应用于实践当中。加深了我对理论知识的理解。**

**批阅者：**

**批阅日期：**

**实验成绩：**

**批注：**