

浙江大学实验报告

一、 实验目的和要求

设计并实现一个精简的图书管理系统，要求具有图书入库、查询、借书、还书、借书证管理等功能。

二、 实验内容和原理

1. 基本数据对象

| 对象名称 | 包含属性 |
|------|---------------------------------------|
| 书 | 书号, 类别, 书名, 出版社, 年份, 作者, 价格, 总藏书量, 库存 |
| 借书证 | 卡号, 姓名, 单位, 类别 (教师 学生等) |
| 管理员 | 管理员 ID, 密码, 姓名, 联系方式 |
| 借书记录 | 卡号, 借书证号 ,借期, 还期, 经手人 (管理员 ID) |

2. 基本功能模块

| 模块名称 | 功能描述 |
|-------|---|
| 管理员登陆 | 输入管理员 ID, 密码; 登入系统 或 返回 ID/密码 错误 |
| 图书入库 | 1.单本入库 2.批量入库 |
| 图书查询 | 要求可以对书的 类别, 书名, 出版社, 年份区间, 作者, 价格区间进行查询 |
| 借书 | 1.输入借书证卡号显示该借书证所有已借书籍 2.输入书号, 如果该书还有库存, 则借书成功, 同时库存数减一 |
| 还书 | 1.输入借书证卡号显示该借书证所有已借书籍 2.输入书号如果该书在已借书籍列表内, 则还书成功, 否则报错 |
| 借书证管理 | 增加或删除一个借书证 |

3. 用户界面

采用图形界面或字符界面。

三、 主要仪器设备

1. 操作系统：Windows 2008
2. 数据库管理系统：SQL Server 2008
3. 图形用户界面应用程序开发框架：QT

四、 操作方法与实验步骤

1. 建立数据库

```

create table admin
(ID char(20), password char(20), name varchar(20), tel char(20), primary key(ID));

create table book
(bno char(20), category char(20), title varchar(20), press varchar(20), year int, author varchar(20),
price decimal(7,2), total int , stock int, primary key(bno), check(stock>=0));

create table card
(cno char(20), name varchar(20), department varchar(20), type char(1), primary key(cno),
check(type in ('T', 'S', 'O')));

create table borrow
(cno char(20), bno char(20), borrow_date datetime, return_date datetime,
brokerage char(20), foreign key(cno)references card,
foreign key(bno)references book, foreign key(brokerage)references );

```

2. 连接数据库

```

QSqlDatabase db = QSqlDatabase::addDatabase("QODBC");
db.setDatabaseName("DRIVER={SQLSERVER};SERVER=(local); DATABASE=library;");
db.open(); QString s="select*from book;";    // s 是查询语句
QSqlQueryModel *m=new QSqlQueryModel;
m->setQuery(s);

```

以上代码可以连接数据库并进行执行查询语句 s。

3. QT 显示查询结果

```

QDialog *d = new QDialog; QTableView *v = new QTableView;
QHBoxLayout *l = new QHBoxLayout;   QSqlQueryModel *m=new QSqlQueryModel;
QString s="select*from book";
l->addWidget(v); d->setLayout(l); m->setQuery(s);
v->setModel(m); d->adjustSize(); d->show();

```

通过以上代码可以把 s 的查询结果在 QT 中以表格的方式显示出来。

4. 各个功能模块的实现

(1) 管理员登陆

```
s=QString("select*from admin where user=\'%1\' and password=\'%2\'");arg(ui->lineEdituser->text()).arg(ui->lineEditpassword->text()); m->setQuery(s);
```

通过判断 `m->rowCount()` 是否等于零，如果 `m->rowCount()` 等于代表没有账号或密码错误，否则登入成功。

(2) 图书入库

```
s=QString("select*from book where bno=\'%1\'").arg(ui->lineEditbno->text());
m->setQuery(s); if(m->rowCount()==0)
{s=QString("insert into book values(...)");m->setQuery(s);}else
{s=QString("update book set stock=stock+%1,total=total+%2 where bno=\'%3\'")...;
m->setQuery(s);} //其中...是省略了的部分
```

图书入库要判断是否已有该图书。如果已经有该图书，则更新书本的总数和库存；否则插入该书。

批量入库时通过以上代码打开存是 `ui->lineEditaddress->text()` 中的文件。

(3) 图书查询

图书查询直接执行语句“`select*from book where ...;`”。如果按要求查询，则 `where` 里输入相应条件；否则可以省略 `where`。

(4) 借书和还书

```
s=QString("select*from book where bno=\'%1\' and stock>0;")...; m->setQuery(s);
if(m->rowCount()==0) QMessageBox::about(NULL, "图书馆", "没有库存!"); else {
s=QString("insert into borrow values(...)"); m->setQuery(s);
s=QString("update book set stock=stock-1 where bno=\'%1\'")...; m->setQuery(s);
QMessageBox::about(NULL, "图书馆", "借书成功!");}
```

借书前要先判断该图书是否有库存。如果有库存，则在 `borrow` 中插入借书

记录并把该图书的库存减一；如果没有库存，则拒绝借书。借书时用 SQL 中的 GETDATE()函数直接获得系统的时间。

```
s=QString("select*from borrow where ... and return_date is null")...; m->setQuery(s);
if(m->rowCount()!=0)QMessageBox::about(NULL, "图书馆", "没有借书记录!");else
{s=QString("update borrow set return_date=GETDATE() where... ")...; m->setQuery(s);
s=QString("update book set stock=stock+1 where bno='%1\\'")...;
QMessageBox::about(NULL, "图书馆", "还书成功!");}
```

还书前要先判断该借书证是否已借该书籍并且没还，return_date 是空代表没还。如果该书籍没还，则更新还书日期并该书库存加一；否则报错。

查看借书记录直接执行语句“select*from borrow where cno=...”即可。

(5)借书证管理

新建借书证直接报行语句“insert into card values(...)”即可。

```
s=QString("select*from borrow where...and return_date is null")...; m->setQuery(s);
if(m->rowCount()!=0)QMessageBox::about(NULL, "图书馆", "有书本没还!"); else {
s=QString("delete from borrow where cno='%1\\'");...;m->setQuery(s);
s=QString("delete from card where cno='%1\\'");...;m->setQuery(s);
QMessageBox::about(NULL, "图书馆", "注销借书证成功!");}
```

注销借书证时首先要判断该借书是否有还没有还的书籍。如果 borrow 有 return_date 为空的记录，代表该借书证有没有还的书。如果所有书都还了，则删除所有借书记录和借书证。

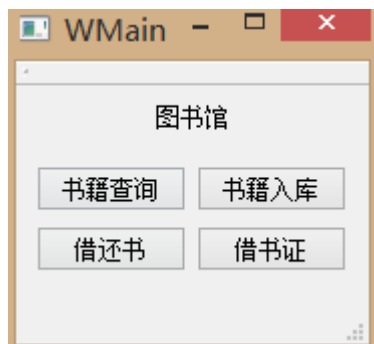
五、 实验结果与分析

1. 管理员登陆



输入帐号和密码，如何帐号和密码正确，则会跳到主界面。

2. 主界面



主界面中有四个按钮，按按钮会跳到对应界面。

3. 图书入库



输入相应信息并点单本入库可以插入单本书籍。输入地址并点批量入库可以把地址的文件中的书籍行进入库。

4. 图书查询



The dialog box is titled "Dialog" and contains a sub-header "图书馆-书籍查询". It features six input fields: "类别" (Category), "书名" (Book Name), "出版社" (Publisher), "作者" (Author), "年份" (Year) with a separator, and "价格" (Price) with a separator. At the bottom, there are three buttons: "查询所有" (Query All), "按条件查询" (Query by Conditions), and "返回" (Return).

点击查询所有按钮可以看到所有的图书，输入相应条件并点按条件查询可以根据输入的条件进行查询。

5. 借书和还书



The dialog box is titled "Dialog" and contains a sub-header "图书馆-借还书". It features two input fields: "卡号" (Card Number) and "书号" (Book Number). Below these are four buttons: "借书" (Borrow Book), "还书" (Return Book), "借书记录" (Borrow Record), and "返回" (Return).

输入卡号和书号并点击相应按钮可以进行借书和还书，输入卡号并点击借书记录可以看到该借书证的借书记录。

6. 借书证管理

Dialog

图书馆-借书证

卡号 姓名

单位 类别

输入相应信息可以新建借书证，输入卡号并点击注销借书证可以注销相应借书证。

六、 讨论、心得

本实验的主要难点在于如何建立一个用户界面，以及如何将用户界面和数据库进行连接。本实验需要自己到网上学习一些关于用户界面软件的应用，所以我花了比较多的时间。虽然本实验难度较大，但是通过本实验不仅能加深 SQL 的理解，还能学会如何做一个图书管理系统以及如何使用 QT。所以本实验大大提高了我的 SQL 应用能力，本次实验是一个很好的学习机会。