

实验 5 数据库程序设计

一、实验目的

- 1、设计并实现一个精简的图书管理系统，具有入库、查询、借书、还书、借书证管理等基本功能。
- 2、通过本次设计来加深对数据库的了解和使用，同时提高自身的系统编程能力。

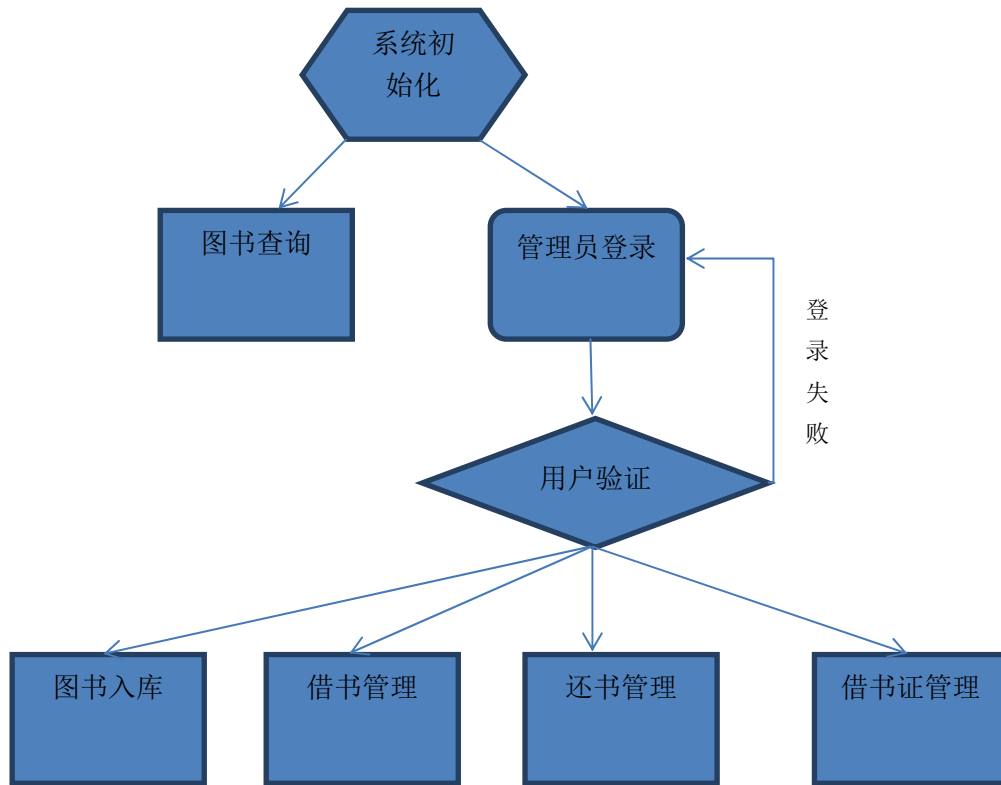
二、实验平台

开发工具：Delphi；数据库平台：SQLServer 2008 Express i

三、总体设计

1、系统架构描述

本系统主要包括管理员登录、图书入库、图书查询、借书管理、还书管理、借书证管理六大功能模块。系统处理基本流程如下：



系统初始时仅有图书查询和管理员登录两个选项，图书查询为公共功能模块，不需要登录也可操作。管理员成功登录后，便可以选择进入图书入库、借书管理、还书管理、借书证管理功能。

各个功能模块说明如下：

模块名称	功能描述
管理员登陆	根据管理员 ID 和密码登录系统
图书入库	1. 单本入库，直接从程序界面上输入 2. 批量入库，从文本文件中批量导入图书数据

图书查询	1. 按书的类别, 书名, 出版社, 年份, 作者, 价格进行查询. 2. 可以点击标题来对相应的字段进行排序
借书	1.输入借书证卡号, 自动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, 则提示借书成功, 同时库存数减 1, 否则输出该书无库存。
还书	1.输入借书证卡号, 自动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, 同时库存加 1. 否则输出出错信息。
借书证管理	增加或删除借书证.

2、数据库表设计

图书信息表(Books)

字段名	数据类型	主键	说明
BookNo	Nvarchar(50)	是	书号
BookType	Nvarchar(50)		图书类别
BookName	Nvarchar(50)		书名
Publisher	Nvarchar(50)		出版社
Year	int		出版年份
Author	Nvarchar(50)		作者
Pirce	money		图书单价
Total	int		总藏书量
Storage	int		库存数
UpdateTime	datetime		添加时间

借书证表(LibraryCard)

字段名	数据类型	主键	说明
CardNo	Nvarchar(50)	是	卡号
Name	Nvarchar(50)		姓名
Department	Nvarchar(50)		单位
CardType	Nvarchar(50)		类别
UpdateTime	datetime		添加时间

用户表(Users)

字段名	数据类型	主键	说明
UserID	Nvarchar(50)	是	用户名
Password	Nvarchar(50)		密码
Name	Nvarchar(50)		姓名
Contact	Nvarchar(50)		联系方式

借书记录表(LibraryRecords)

字段名	数据类型	主键	说明
-----	------	----	----

FID	int	是	标识列
CardNo	Nvarchar(50)		借书卡号
BookNo	Nvarchar(50)		书号
LentDate	datetime		借书日期
ReturnDate	datetime		还书日期
Operator	Nvarchar(50)		经手人（管理员 ID）

3、所用开发技术

本次开发所用技术可以从三个方面来说明：SQLServer，Delphi，ADO。

SQL Server 是由 Microsoft 开发和推广的关系数据库管理系统（DBMS），它所采用的是客户机/服务器体系结构，具有良好的可伸缩性，同时也有图形化的用户管理界面，使得管理数据库更为直观及方便。在 SQLServer 中创建数据库和表，可以直接在图形化表格工具里录入和修改，而不必每次都直接通过 SQL 语言操作。数据表创建完成后，亦可直接编辑表中的数据，方便了测试。同时，它也有一个完好的 SQL 编辑器，可以自动检测语法错误，写好后也可以直接执行 SQL 语句。在把 SQL 语句集成到程序之前，可以先在 SQL 编辑器里编写执行，检验无误后，再在程序中应用，这样就避免了频繁在程序中查找 SQL 语言错误，提高了工作效率。本次开发中所使用的是 SQLServer 2008 Express 版本，数据库表的创建等在他的图形化管理工具中完成。

本次程序编写上，使用的是 Delphi 开发工具。Delphi 是一种基于 Object Pascal 语言的面向对象的软件开发平台，它拥有极快的编译器，并提供了丰富的组件集、强大的代码自动生成功能和数据库管理工具等。使用它的集成开发环境，编程人员可以简单快速地开发各种应用程序，许多传统、常规的编程都可以借助类库来实现。在数据库和网络处理功能开发方面，Delphi 提供了迅速和简洁的方法，因此，在这方面它是理想的软件开发平台。

对于访问数据库，Delphi 主要提供了 BDE 和 ADO 等组件，可以十分方便和快捷地访问和操作远程或本机数据库。ADO 是微软公司提供的用于存储数据源的 COM 组件，它提供了编程语言和统一数据访问方式 OLE DB 的一个中间层，允许开发人员编写访问数据的代码而不用关心数据库是如何实现的，而只关心到数据库的连接。通过 ADO 不仅能访问基于 SQLServer 的数据库，而且可以通过 ODBC 访问所有支持 ODBC 的数据库。Delphi 中的 ADO 组件封装了 ADO 对象，可用于建立数据库应用程序。使用 ADO 还有一个重要优势，就是它被内置于微软所有的 Windows 操作系统中，所以在绝大多数情况下，使用 ADO 技术不需要在每一台计算机中重新安装 ADO。

本次的 ADO 应用程序开发中，所用到的几个主要组件有 TADOConnection、TADOQuery。

TADOConnection 组件用于建立与 ADO 数据库之间的联系，各种 ADO 数据集及操作组件可以共用 TADOConnection 连接来执行命令以及读取数据。它的主要属性有 ConnectionString，指明了数据库的连接信息，在设计程序时可以通过可视化工具来创建和测试连接字符串；Connected 属性则指明了一个与数据库的联系是否被激活，当其值为 True 时，就建立了一个与 ADO 数据库之间的联系。

TADOQuery 组件用于执行 SQL 语句，包括 SELECT、INSERT、UPDATE、DELETE 等。使用前，

必须为它设置数据库连接，可以直接通过 `ConnectionString` 属性来设置，也可以通过 `Connection` 属性来指定 `TADOConnection` 对象以共享数据连接。成功连接上数据库后，可以修改它的 `SQL` 属性来指明所要执行的 SQL 语句，然后通过 `Open` 或 `ExecuteSQL` 方法来执行。当执行的是查询语句时，会返回查询到的数据集，此时就可以通过 `Next()` 方法来遍历数据集，以及通过 `Fields` 属性取得字段值。

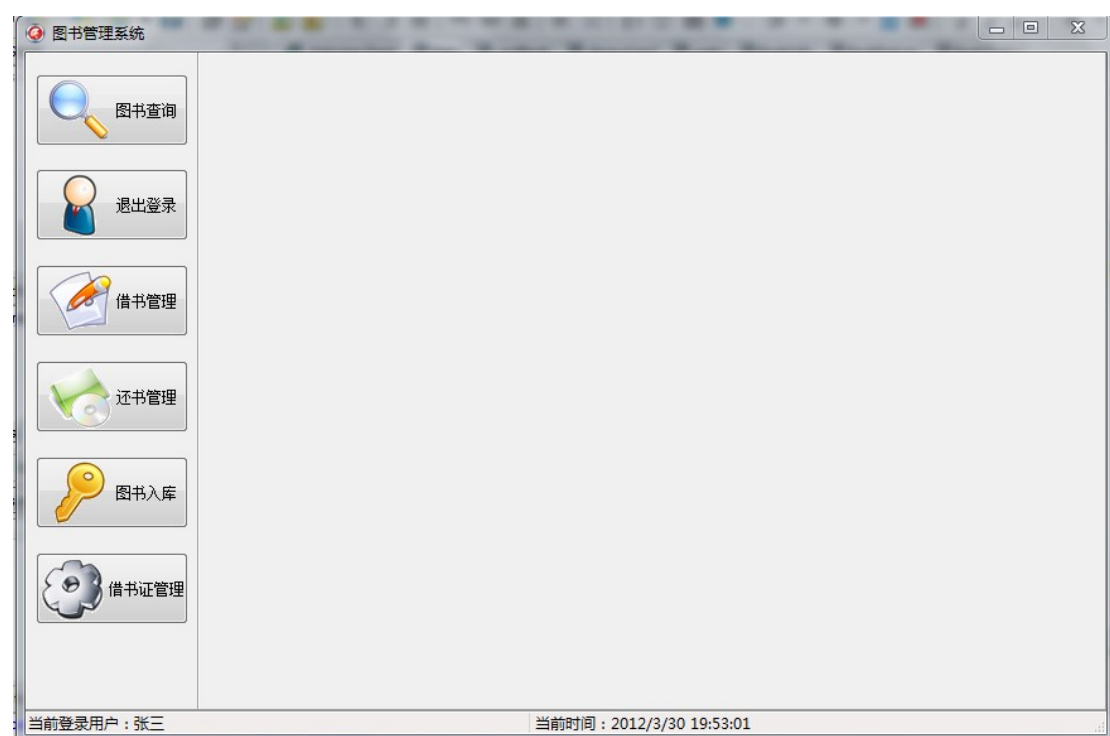
4、成员任务描述

本次从数据库设计，到各个模块的功能编写，皆为一入独立完成。程序模块设计上主要参考了实验指导书的说明，结合自己的理解来理清思路，在明确要完成的目标后，便收集和查阅了一些技术资料来学习，了解了 Delphi 中 ADO 组件的使用及 `SQLServer` 工具的使用，最后选定 `Delphi+ADO+SQLServer` 方案进行开发完成本系统，同时花费了若干小时来安装配置开发环境。

四、详细设计

1、主界面设计

程序界面采用类似于多标签页的方式工作，左边是导航栏，点击相应的按钮时，将在右边打开对应的窗体。主界面的功能主要集中在导航上，而并不过多关注具体的图书查询等模块的功能实现，在具体编写上表现为主界面是单独的窗体，其他的各个模块也在单独的窗体中，这样也方便把不同人所写的窗体模块集成到一起。如图所示：



具体的实现采用了 Dock 窗体技术，把一个窗体停靠到另一个窗体中，在 Delphi 中可以方便地通过 `ManualDock` 过程实现。以下是代码实现片断：

```
procedure TMainForm.btnAddClick(Sender: TObject);
begin
    BookForm.BorderStyle := bsNone; //设置窗体为无边框
    BookForm.Align := alClient; //设置窗体对齐方式，alClient 为充满整个客户区
```

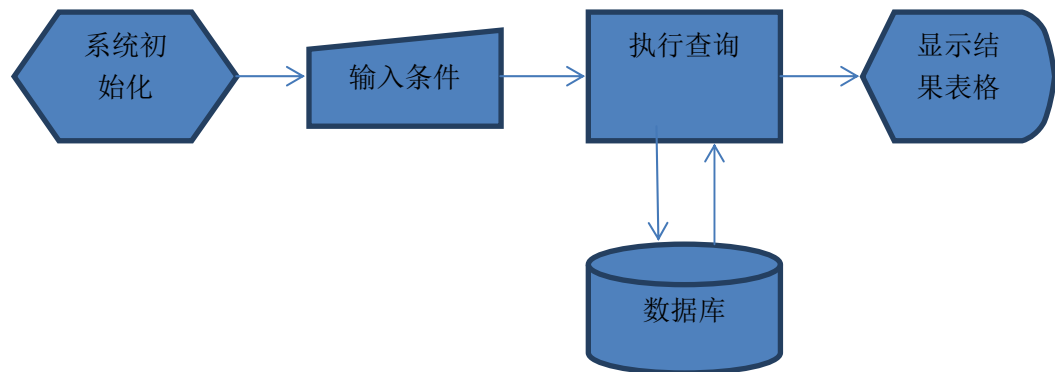
```

BookForm.ManualDock(Panel2); //把 BoookForm 停靠到 Panel2 组件之上
BookForm.Show; //显示 BookForm 窗体
Caption := '图书管理系统-[图书入库]'; //设置标题，方便用户知道自己当前所在的模块
end;

```

2、图书查询模块

图书查询不需要登录，在系统初始化完成后可直接开始使用，其基本工作流程是：输入查询条件→组合查询条件→执行查询→返回结果。如下图：



查询界面设计如下：

书号	类别	书名	出版社	年份	作者	价格	总藏书量	库存
9787115145543	编程语言	C++ Primer 中文版	人民邮电出版社	2006	(美) Stanley E.	99	30	30
9787302224464	编程语言	C程序设计 (第4版)	清华大学出版社	2010	谭浩强	29	10	10
9787111128069	编程语言	C程序设计语言 (第2版)	机械工业出版社	2004	(美) 克尼汉, 莫里森	30	10	10
9787115264480	数据库	Hadoop 实战	人民邮电出版社	2011	(美) Chuck La	59	1	0
9787111213826	编程语言	Java 编程思想 (第4版)	机械工业出版社	2007	(美) 埃克尔斯	108	30	30
9787115244857	数据库	Oracle Database 9i/10g 入门指南	人民邮电出版社	2011	(美) 凯特 著	109	2	2
9787115179289	编程语言	编程珠玑 (第2版)	人民邮电出版社	2008	Jon Bentley 著	39	20	20
9787111205388	数据库	数据挖掘概念与技术	机械工业出版社	2007	(加) 韩家炜, 张	55	8	8

查询的条件有书名、类别、作者、出版社、年份区间、价格范围，当对应的条件被输入时，就要根据这个条件来进行查询，如果没有填写，则忽略掉。具体实现上是，逐个判断每个输入框的数据，如果是空，则不作处理，如果非空，则合成一个新的查询条件，和之前的 SQL 语句连接在一起，最后执行查询。

主要功能代码如下：

```

var
  sql: string;
begin
  sql := 'SELECT * FROM Books WHERE 1=1'; //构造查询语句
  //当书名不为空时，组合书名作查询条件
  if edtBookName.Text <> '' then
    sql := sql + ' AND BookName LIKE "%' + Trim(edtBookName.Text) + '%"';
  //当类别不为空时，组合类别作查询条件，以下略同
  if cbBookType.Text <> '' then
    sql := sql + ' AND BookType=' + QuotedStr(Trim(cbBookType.Text));
  if edtAuthor.Text <> '' then
    sql := sql + ' AND Author LIKE "%' + Trim(edtAuthor.Text) + '%"';
  if cbPublisher.Text <> '' then
    sql := sql + ' AND Publisher=' + QuotedStr(Trim(cbPublisher.Text));
  if (cbYear1.Text <> '') and (cbYear2.Text <> '') then
    begin
      sql := sql + ' AND Year>=' + cbYear1.Text + ' AND Year<=' + cbYear2.Text;
    end;
  if (edtPrice1.Text <> '') and (edtPrice2.Text <> '') then
    begin
      sql := sql + ' AND Price >= ' + edtPrice1.Text + ' AND Price <= ' + edtPrice2.Text;
    end;
  sql := sql + ' ORDER BY BookName'; //默认按书名进行排序
  //执行查询
  ADOQuery1.Close;
  ADOQuery1.SQL.Text := sql;
  ADOQuery1.Open;

```

结果排序功能，一开始使用一个输入来选择要按哪个字段进行排序，经过考虑后，修改为点击对应的标题头时自动排序，这也符合大多数软件的操作习惯。在 Delphi 中实现，主要是通过 TADOQuery 组件的 Sort 属性来完成，先获取当前点击的字段，然后用此字段来设置 Sort 属性。代码如下：

```

procedure TQueryForm.DBGrid1TitleClick(Column: TColumn);
begin
  if ADOQuery1.Sort <> (Column.FieldName + ' ASC') then
    ADOQuery1.Sort := Column.FieldName + ' ASC'
  else
    ADOQuery1.Sort := Column.FieldName + ' DESC';
end;

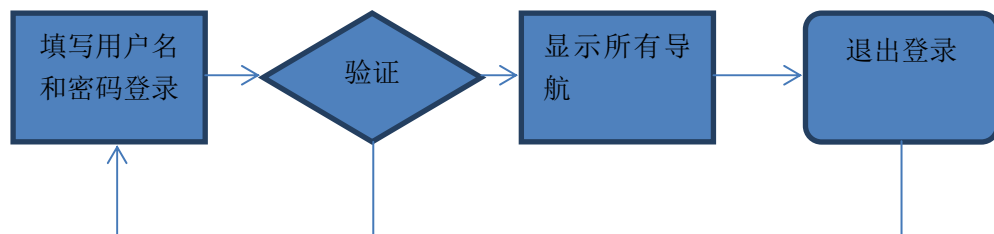
```

3、管理员登录模块

本模块主要实现了管理员根据用户名和密码登录系统，在默认没有登录的情况下，程序主界面上并不会显示出图书入库、借书管理、还书管理等模块的导航按钮，当管理员成功登录后，所有功能导航按钮将显现，然后可以进入操作。登录界面如图：

The image shows a Windows-style dialog box titled "管理员登录" (Administrator Login). It contains two text input fields: "用户名:" (Username) and "密码:" (Password). Below the fields are two buttons: a green "登录" (Login) button with a checkmark icon, and a red "取消" (Cancel) button with an 'X' icon.

流程基本如下:



校验部分主要通过用户的输入，组合出对应的 SQL 语句，然后进行查询，如果查询没有结果，则登录失败，否则登录成功，代码如下：

```
ADOQuery1.SQL.Text := 'SELECT UserID, Name FROM Users WHERE UserID=:UserID AND Password=:Password'; //建立动态 SQL 语句
```

```
ADOQuery1.Prepared := True;
```

```
ADOQuery1.Parameters.ParamByName('UserID').Value := edtName.Text; //给参数赋值
```

```
ADOQuery1.Parameters.ParamByName('Password').Value := edtPassword.Text; //给参数赋值
```

```
ADOQuery1.Open; //执行查询
```

```
//判断查询结果是否为空
```

```
if ADOQuery1.IsEmpty then
```

```
begin
```

```
    Application.MessageBox('用户名或密码不正确', '提示');
```

```
    ModalResult := mrNone;
```

```
    Exit;
```

```
end;
```

如果登录成功，则显示所有功能导航，并重设按钮标题，如下：

```
procedure TMainForm.DoLogin;
```

```
begin
```

```
    btnLent.Show; //显示借书管理
```

```
    btnReturn.Show; //显示还书管理
```

```
    btnAdd.Show; //显示图书入库
```

```
    btnCardMgr.Show; //显示借书证管理
```

```
    btnLogin.Caption := '退出登录';  
end;
```

在退出登录时，则要关闭所有打开的管理模块，并隐藏功能导航，如下：

```
procedure TMainForm.DoLogout;  
begin  
    //隐藏导航按钮  
    btnLent.Hide;  
    btnReturn.Hide;  
    btnAdd.Hide;  
    btnCardMgr.Hide;  
    btnLogin.Caption := '管理员登录';  
  
    //清空内部登录状态  
    CurrentUserName := '';  
    CurrentUserID := '';  
    StatusBar1.Panels[0].Text := '当前登录用户：';  
    Caption := '图书管理系统';  
  
    //关闭所有管理模块  
    LentForm.Close;  
    ReturnForm.Close;  
    BookForm.Close;  
    CardForm.Close;  
    QueryForm.Close;  
end;
```

4、图书入库模块

本模块主要实现了图书信息的录入，以及批量导入功能。管理员可以在软件界面上逐条录入图书的各项信息，然后点击添加按钮，就会把当前录入的图书信息保存到数据库中。如果现有大量数据，则可选择从文本文件中导入，格式是：书号, 类别, 书名, 出版社, 年份, 作者, 价格, 数量。每项之间用英文逗号分隔开来，每行一条记录。程序导入时会自动检测是否符合格式，并忽略掉不符合格式的数据行。界面如图：

图书管理系统-[图书入库]

图书查询 退出登录 借书管理 还书管理 图书入库 借书证管理

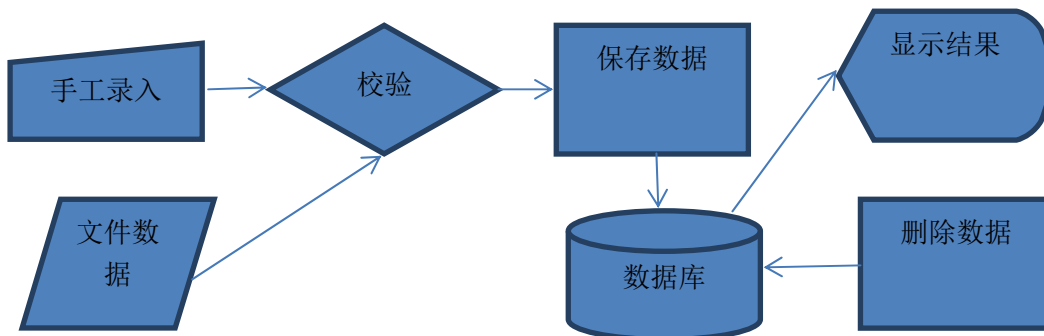
书号 类别 书名 出版社 年份 作者 价格 数量

添加 删除 导入

书号	类别	书名	出版社	年份	作者	价格	数量
9787111128069	编程语言	C程序设计语言(第2版)	机械工业出版社	2004	(美) 克尼茨	30	10
9787111205388	数据库	数据挖掘概念与技术	机械工业出版社	2007	(加) 韩家炜	55	8
9787111213826	编程语言	Java编程思想(第4版)	机械工业出版社	2007	(美) 埃克尔	108	30
9787115145543	编程语言	C++ Primer 中文版(第2版)	人民邮电出版社	2006	(美) Stanle	99	30
9787115179289	编程语言	编程珠玑(第2版)	人民邮电出版社	2008	Jon Bentley 著	39	20
9787115244857	数据库	Oracle Database 9i/10g 入门与进阶	人民邮电出版社	2011	(美) 凯特	109	2
9787115264480	数据库	Hadoop实战	人民邮电出版社	2011	(美) Chuck	59	1
9787302224464	编程语言	C程序设计(第4版)	清华大学出版社	2010	谭浩强	29	10

当前登录用户: 张三 当前时间: 2012/3/30 21:12:52

基本工作流程为:



在界面初次显示时,通过查询语句自动显示当前数据库中的图书数据,按数据更新时间排序显示,当录入新的数据时,将自动更新显示当前录入的数据。录入数据的校验部分,主要是检查录入的结果是否完整,以及录入的价格和数字是否为合法的数字。完整性检查方面,因为控件比较多,所以通过以下标遍历控件的方式来检查。对于数量字段,已限定文本框只能输入数字,所以不必再作多余检测。对于价格,则通过捕获 StrToFloat 函数的异常来检测。

校验通过后,数据的插入主要通过 ADO 组件的 Insert 和 Post 方法来完成提交,代码如下:

```

ADOQuery1.Insert;
ADOQuery1.FieldName('BookNo').AsString := Trim(edtBookNo.Text);
ADOQuery1.FieldName('BookType').AsString := Trim(cbBookType.Text);
ADOQuery1.FieldName('BookName').AsString := Trim(edtBookName.Text);
ADOQuery1.FieldName('Publisher').AsString := Trim(cbPublisher.Text);
ADOQuery1.FieldName('Year').AsInteger := StrToInt(cbYear.Text);
ADOQuery1.FieldName('Author').AsString := Trim(edtAuthor.Text);

```

```

ADOQuery1.FieldByName('Price').AsFloat := StrToFloat(edtPrice.Text);
ADOQuery1.FieldByName('Total').AsInteger := StrToInt(edtStorage.Text);
ADOQuery1.FieldByName('Storage').AsInteger := StrToInt(edtStorage.Text);
ADOQuery1.FieldByName('UpdateTime').AsDateTime := Now;
ADOQuery1.Post;

```

批量导入时，通过 TStringList 一次性加载文本文件，这样文件内容就会自动装载到一个字符串列表中，方便处理。在进行字符串的分解时，也使用了另一个 TStringList 变量，它可以通过 Delimiter 属性来指定用于分割的字符，然后把待分割的字符串赋值到它的 DelimitedText 属性，TStringList 就会自动把字符串分割成若干项。导入部分代码如下：

```

procedure TBookForm.ImportFromFile(FileName: string);

```

```

var

```

```

    list, split: TStringList;

```

```

    str: string;

```

```

begin

```

```

    list := TStringList.Create;

```

```

    split := TStringList.Create;

```

```

    split.Delimiter := ','; //指定分隔字符

```

```

    split.StrictDelimiter := True; //指定严格分割模式

```

```

    list.LoadFromFile(FileName); //加载文本文件

```

```

    //遍历文件的每一行

```

```

    for str in list do

```

```

    begin

```

```

        split.DelimitedText := str; //分割字符串

```

```

        if split.Count <> 8 then //判断数据项数是否正确

```

```

            Continue;

```

```

        Try //判断数字部分是否正确

```

```

            StrToFloat(split[6]);

```

```

            StrToInt(split[7]);

```

```

            StrToInt(split[4])

```

```

        except

```

```

            Continue;

```

```

        end;

```

```

    //插入数据

```

```

    ADOQuery1.Insert;

```

```

    ADOQuery1.FieldByName('BookNo').AsString := Trim(split[0]);

```

```

    ADOQuery1.FieldByName('BookType').AsString := Trim(split[1]);

```

```

    ADOQuery1.FieldByName('BookName').AsString := Trim(split[2]);

```

```

    ADOQuery1.FieldByName('Publisher').AsString := Trim(split[3]);

```

```

    ADOQuery1.FieldByName('Year').AsInteger := StrToInt(split[4]);

```

```

    ADOQuery1.FieldByName('Author').AsString := Trim(split[5]);

```

```

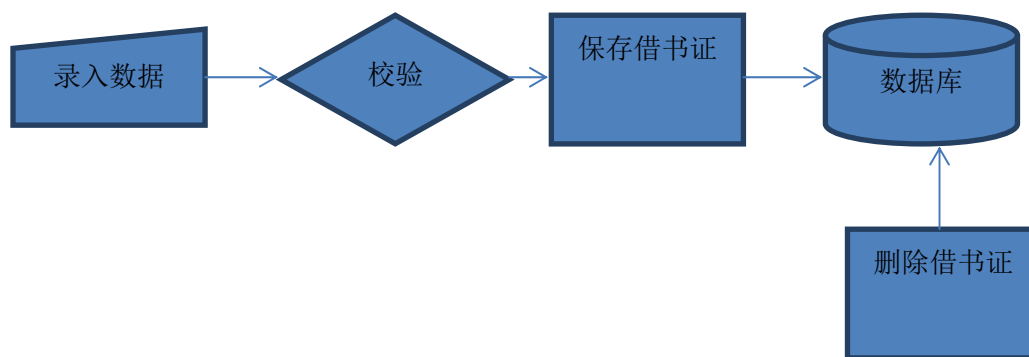
ADOQuery1.FieldByName('Price').AsFloat := StrToFloat(split[6]);
ADOQuery1.FieldByName('Total').AsInteger := StrToInt(split[7]);
ADOQuery1.FieldByName('Storage').AsInteger := StrToInt(split[7]);
ADOQuery1.FieldByName('UpdateTime').AsDateTime := Now;
ADOQuery1.Post;
end;

split.Free;
list.Free;
end;

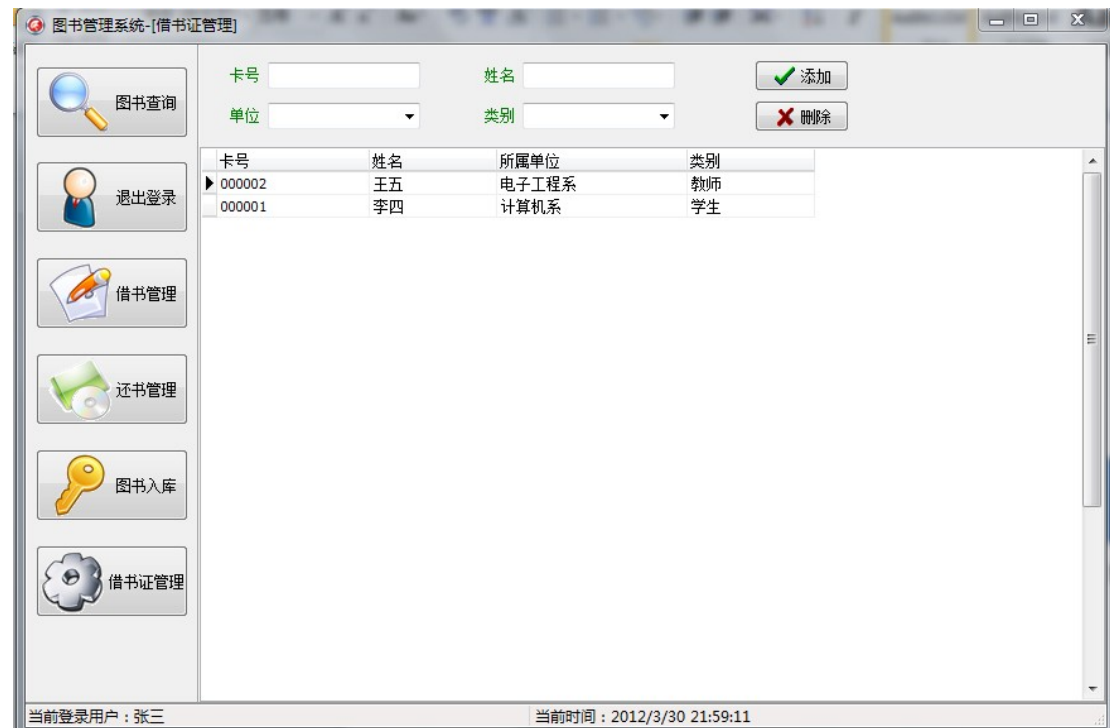
```

5、借书证管理模块

本模块主要实现了借书证的添加和删除，其工作基本流程为：



当输入数据后，首先对数据进行完整性校验，通过后再把数据插入到数据库中。当删除借书证时，则先询问是否确定删除，确认后再执行删除操作。界面如下：



添加记录实现代码如下：

```
procedure TCardForm.btnAddClick(Sender: TObject);
begin
    //完整性校验
    if (edtName.Text = '') or (edtCardNo.Text = '') or (cbDepartment.Text = '') or (cbCardType.Text = '')
then
    begin
        Application.MessageBox('请把信息填写完整', '提示');
        Exit;
    end;

    //插入数据到表中
    ADOQuery1.Insert;
    ADOQuery1.FieldByName('CardNo').AsString := Trim(edtCardNo.Text);
    ADOQuery1.FieldByName('Name').AsString := Trim(edtName.Text);
    ADOQuery1.FieldByName('Department').AsString := Trim(cbDepartment.Text);
    ADOQuery1.FieldByName('CardType').AsString := Trim(cbCardType.Text);
    ADOQuery1.FieldByName('UpdateTime').AsDateTime := Now;
    ADOQuery1.Post;
end;
```

删除部分实现代码如下：

```
procedure TCardForm.btnDeleteClick(Sender: TObject);
var
    text: string;
begin
    if ADOQuery1.IsEmpty then
        Exit;
    text := Format('确定要删除借书证%s 吗？ ', [ADOQuery1.FieldByName('CardNo').AsString]);
    if Application.MessageBox(PChar(text), '提示', MB_YESNO or MB_ICONQUESTION) = IDYES
then
        ADOQuery1.Delete;
end;
```

6、借书管理模块

本模块主要实现了查询对应借书证借书记录和借书登记功能。当管理员在借书证卡号里录入数据时，会自动执行查询，如果借书证卡号存在，将在下面的表格中自动显示出该借书证所借的图书信息。录入书号后，点击借出按钮，将把该书的借出信息添加到 LibraryRecords 表中，同时 Books 表中的库存里自动减 1。界面如图：

更新库存部分部分代码实现如下：

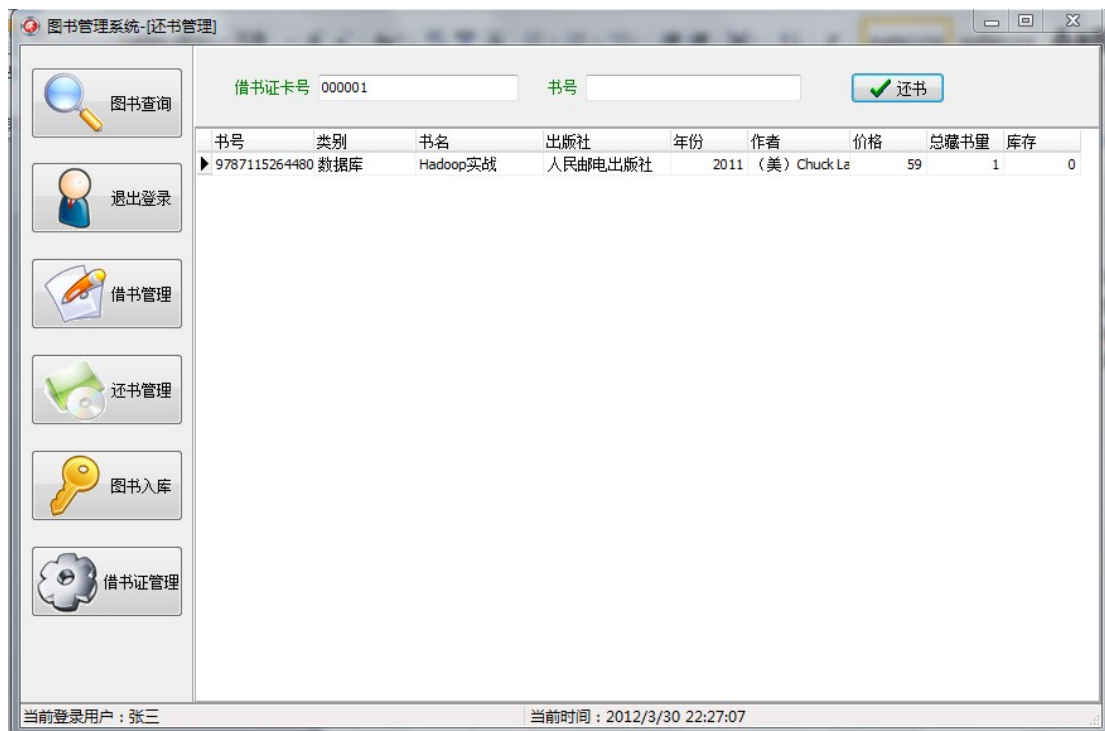
```
storage := ADOQuery2.Fields[0].AsInteger;  
if storage > 0 then //库存数大于 0 时才能外借  
begin  
    ADOQuery2.Edit;  
    ADOQuery2.Fields[0].AsInteger := storage - 1; //库存数减 1  
    ADOQuery2.Post;  
    Lent;  
    Query(edtCardNo.Text);  
    Application.MessageBox('借书成功', '提示');  
End
```

查询借书证所借的图书信息部分实现代码如下，在这里只显示该借书证未还的图书信息，已还的不再显示，当图书未归还时，其对应的图书记录的归还日期字段值为空，所以此处要作 NULL 值的判断：

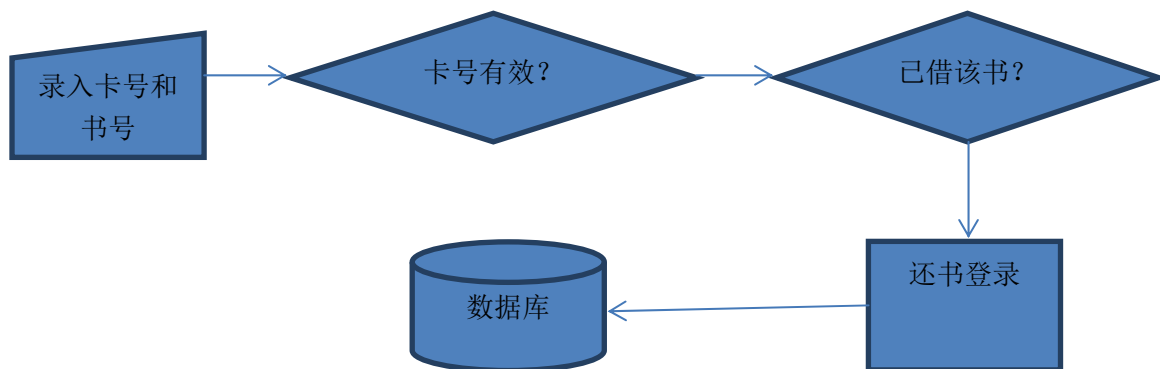
```
procedure TLentForm.Query(CardNo: string);  
begin  
    if CardNo = '' then  
        Exit;  
    ADOQuery1.Close;  
    ADOQuery1.SQL.Text := 'SELECT * FROM Books WHERE BookNo in (SELECT BookNo FROM  
LibraryRecords WHERE CardNo=:CardNo AND ReturnDate is null)'; //只显示还未归还的图书  
    ADOQuery1.Prepared := True;  
    ADOQuery1.Parameters[0].Value := CardNo;  
    ADOQuery1.Open;  
end;
```

7、还书管理模块

本模块主要实现了查询对应借书证借书记录和还书登记功能。当管理员在借书证卡号里录入数据时，会自动执行查询，如果借书证卡号存在，将在下面的表格中自动显示出该借书证所借的图书信息。录入书号后，点击还书按钮，将把该书的还书信息更新到 **LibraryRecords** 表中，同时 **Books** 表中的库存里自动加 1。如果并未借出指定的图书，则提示没有借出此图书。界面如图：



基本工作流程为:



查询借书卡借书信息部分的实现代码与借书管理中的相同。判断是否借出指定的图书，通过查询 LibraryRecords 表中借书卡号对应的信息来确定，实现代码如下：

```

ADOQuery2.Close;
ADOQuery2.SQL.Text := 'SELECT * FROM LibraryRecords WHERE CardNo=:CardNo AND BookNo=:BookNo AND ReturnDate is null'; //查询尚未归还的图书
ADOQuery2.Prepared := True;
ADOQuery2.Parameters.ParamByName('CardNo').Value := edtCardNo.Text; //指定借书卡号
ADOQuery2.Parameters.ParamByName('BookNo').Value := edtBookNo.Text; //指定书号
ADOQuery2.Open;

if ADOQuery2.IsEmpty then
begin
  Application.MessageBox('没有借出该图书', '提示');
end
  
```

```
Exit;  
end;
```

当确定输入的书号被借出时，则把 **LibraryRecords** 表中对应的借书记录的 **ReturnDate** 字段更新为当前时间，指明已经归还图书。实现代码如下：

```
ADOQuery2.Edit;  
ADOQuery2.FieldByName('ReturnDate').AsDateTime := Now;  
ADOQuery2.Post;
```

同时，还要把图书的库存数加 1，实现代码如下：

```
ADOQuery2.Close;  
ADOQuery2.SQL.Text := 'SELECT Storage FROM Books WHERE BookNo=:BookNo'; //查询库存  
ADOQuery2.Prepared := True;  
ADOQuery2.Parameters.ParamByName('BookNo').Value := edtBookNo.Text; //指定书号  
ADOQuery2.Open;  
if not ADOQuery2.IsEmpty then  
begin  
    storage := ADOQuery2.Fields[0].Value;  
    ADOQuery2.Edit;  
    ADOQuery2.Fields[0].Value := storage + 1; //库存数加 1  
    ADOQuery2.Post;  
end;
```