**2023-2024学年第一学期**

**图书馆管理系统大作业**

**[一、数据库设计 1](#_Toc22986)**

[1 概念设计 1](#_Toc7774)

[2 逻辑结构设计 2](#_Toc1029)

[3 物理结构设计 3](#_Toc21979)

**[二、 功能实现 6](#_Toc16687)**

[1 登录 6](#_Toc24864)

[2 读者主页 8](#_Toc3637)

[3 借阅图书界面 9](#_Toc2090)

[4 借阅确认界面 11](#_Toc24287)

[5 管理员主页 14](#_Toc29813)

[6 图书管理模块 15](#_Toc28737)

[7 出版社信息管理 16](#_Toc27177)

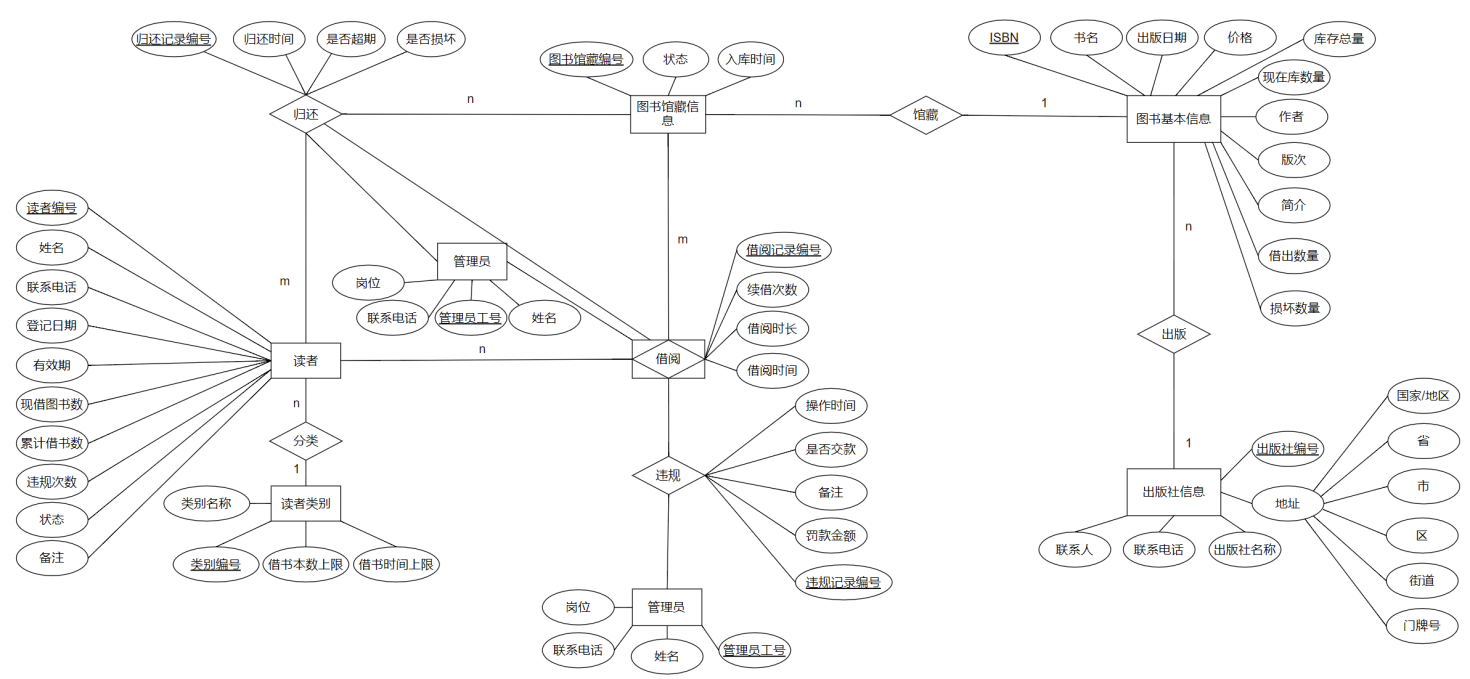
[8 图书基本信息管理 21](#_Toc3004)

# 一、数据库设计

## 1 概念设计

项目背景：某高校图书馆准备建设《图书馆管理系统》，主要用户角色和核心业务分析如下。

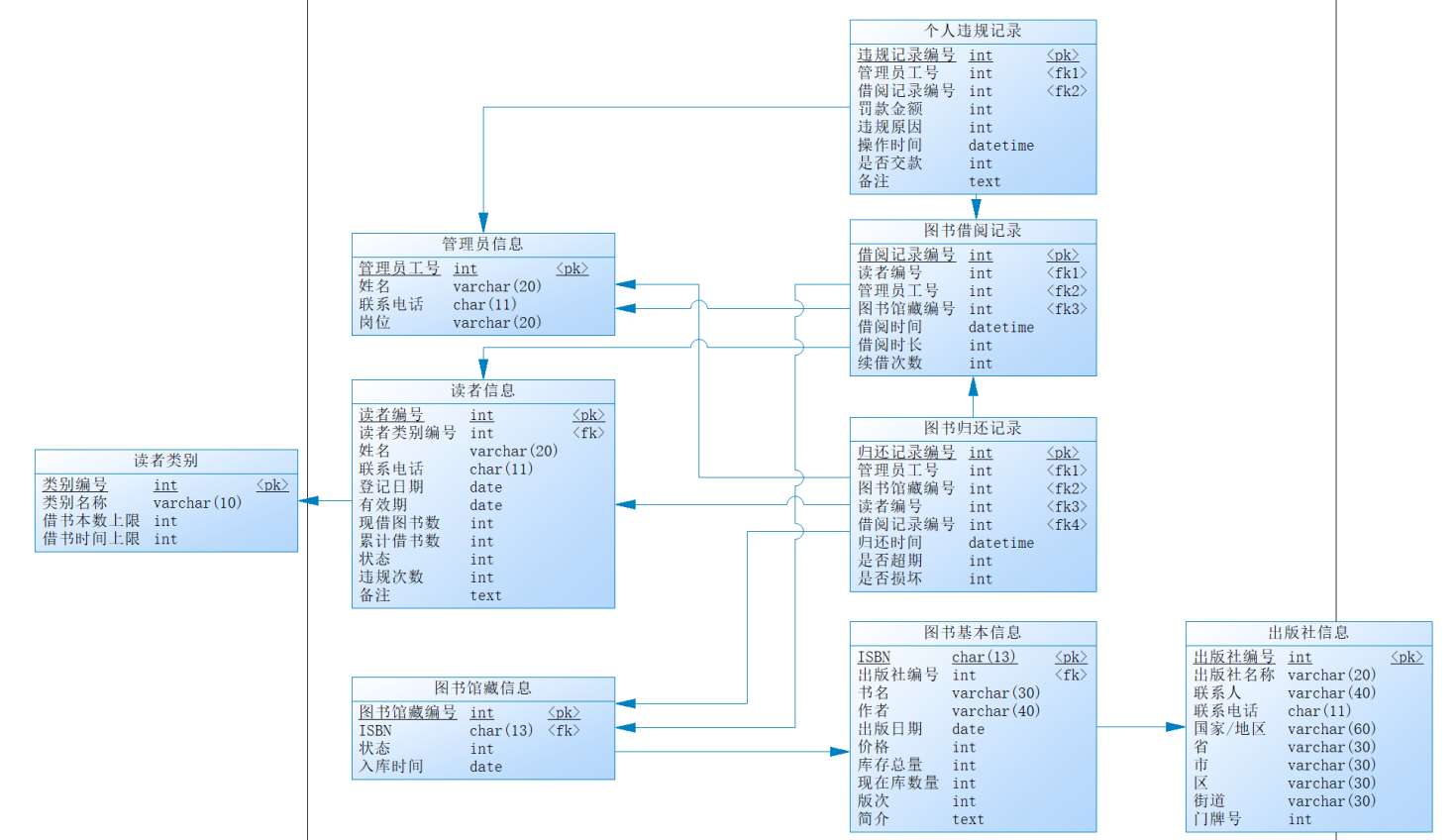
|  |
| --- |
| 用户角色：  1）读者：有三类：  教师：一次能借20本书，能借60天；  职工：一次能借15本书，能借60天；  学生：一次能借10本书，能借30天；  2）管理员：图书馆工作人员，负责办理图书借阅归还等业务。  核心业务：  （1）图书借阅：读者到图书馆借书；  （2）图书归还：读者到图书馆还书；  （3）超期罚款：借书超期未还要罚款，并记入个人违规记录；  （4）图书丢失：读者丢失图书，要赔钱，并记入个人违规记录。 |



## 2 逻辑结构设计

|  |
| --- |
| 管理员（管理员工号，姓名，联系电话，岗位）  读者（读者编号，读者类别编号，姓名，联系电话，登记日期，有效期，现借图书数，累计借书数，状态，违规次数，备注）  外键：读者类别编号  读者类别（类别编号，类别名称，借书本数上限，借书时间上限）  图书馆藏信息（图书馆藏编号，ISBN，状态，入库时间）  外键：ISBN  图书基本信息（ISBN，书名，作者，出版社编号，出版日期，价格，库存总量，现在库数量，借出数量，损坏数量，版次，简介）  外键：出版社编号  出版社信息（出版社编号，出版社名称，联系人，联系电话，国家/地区，省，市，区，街道，门牌号）  图书归还记录（归还记录编号，读者编号，借阅记录编号，图书馆藏编号，管理员工号，归还时间，是否超期，是否损坏）  外键：读者编号，图书馆藏编号，管理员工号，借阅记录编号  图书借阅记录（借阅记录编号，读者编号，图书馆藏编号，管理员工号，借阅时间，借阅时长，续借次数）  外键：读者编号，图书馆藏编号，管理员工号  个人违规记录（违规记录编号，借阅记录编号，违规原因，罚款金额，操作时间，是否交款，备注，管理员工号）  外键：借阅记录编号，管理员工号 |

## 3 物理结构设计



（1）读者信息表

高校图书馆的读者用户包括广大学生、教师和职工。学生读者信息表主要存储高校各个专业和不同类别（本科生、硕士研究生和博士研究生等）的学生基本信息。学生读者信息表如下所示。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| RID | int | 15 | 是 |  | 否 | 读者编号 |
| KID | Int |  |  | 是 | 否 | 读者类别编号 |
| RName | Varchar(20) | 20 |  |  |  | 姓名 |
| RPhone | Char(11) | 11 |  |  |  | 联系电话 |
| RRegister | date |  |  |  |  | 登记日期 |
| RValid | date |  |  |  |  | 有效期 |
| RBorNow | int |  |  |  |  | 现借图书数 |
| RBor | int |  |  |  |  | 累计借书数 |
| RState | int |  |  |  |  | 状态（0为正常，1为挂失，2为注销） |
| RVio | int |  |  |  |  | 违规次数 |
| RComment | text |  |  |  |  | 备注 |
| Password | Varchar(45) |  |  |  | 否 | 密码 |

1. 读者类别表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| KID | Int |  | 是 |  | 否 | 类别编号 |
| KName | Varchar(10) | 10 |  |  |  | 类别名称 |
| Blimit | int |  |  |  |  | 借书本数上限 |
| TLimit | int |  |  |  |  | 借书时间上限 |

（3）管理员信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| MID | Int |  | 是 |  | 否 | 管理员工号 |
| MName | Varchar(20) | 20 |  |  |  | 姓名 |
| MPhone | Char(11) | 11 |  |  |  | 联系电话 |
| MPost | Varchar(20) | 20 |  |  |  | 岗位 |

（4）图书馆藏信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| LID | Int |  | 是 |  | 否 | 图书馆藏编号 |
| ISBN | char(13) | 13 |  | 是 | 否 | ISBN编号 |
| LState | int |  |  |  |  | 状态（0为借出，1为在库） |
| LStore | Date |  |  |  |  | 入库时间 |

（5）图书基本信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| ISBN | Char(13) | 13 | 是 |  | 否 | ISBN编号 |
| PID | int |  |  | 是 | 否 | 出版社编号 |
| BName | Varchar(30) | 30 |  |  |  | 书名 |
| BAuthor | Varchar(40) | 40 |  |  |  | 作者 |
| BPubDate | date |  |  |  |  | 出版日期 |
| BPrice | int |  |  |  |  | 价格 |
| BStorage | int |  |  |  |  | 库存总量 |
| BStorNow | int |  |  |  |  | 现在库数量 |
| BBorNow | Int |  |  |  |  | 借出数量 |
| BDestroy | Int |  |  |  |  | 损坏数量 |
| BVersion | int |  |  |  |  | 版次 |
| BIntro | text |  |  |  |  | 简介 |

（6）出版社信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| PID | Int |  | 是 |  | 否 | 出版社编号 |
| PName | Varchar(20) | 10 |  |  |  | 出版社名称 |
| PContact | Varchar(40) | 40 |  |  |  | 联系人 |
| PPhone | Char(11) | 11 |  |  |  | 联系电话 |
| PCounRe | Varchar(60) | 60 |  |  |  | 国家/地区 |
| PProvince | Varchar(30) | 30 |  |  |  | 省 |
| PCity | Varchar(30) | 30 |  |  |  | 市 |
| PDistinct | Varchar(30) | 30 |  |  |  | 区 |
| PStreet | Varchar(30) | 30 |  |  |  | 街道 |
| PDNo | int |  |  |  |  | 门牌号 |

（7）图书借阅记录表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| BorID | Int |  | 是 |  | 否 | 借阅记录编号 |
| RID | int |  |  | 是 | 否 | 读者编号 |
| MID | int |  |  | 是 | 否 | 管理员工号 |
| LID | int |  |  | 是 | 否 | 图书馆藏编号 |
| BorTime | datetime |  |  |  |  | 借阅时间 |
| BorLen | int |  |  |  |  | 借阅时长 |
| BorCount | int |  |  |  |  | 续借次数 |

（8）图书归还记录表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| ReID | Int |  | 是 |  | 否 | 归还记录编号 |
| MID | Int |  |  | 是 | 否 | 管理员工号 |
| BorID | Int |  |  | 是 |  |  |
| LID | int |  |  | 是 | 否 | 图书馆藏编号 |
| RID | int |  |  | 是 | 否 | 读者编号 |
| RTime | datetime |  |  |  |  | 归还时间 |
| RExceed | int |  |  |  |  | 是否超期（0否，1是） |
| RDestroy | int |  |  |  |  | 是否损坏（0否，1是） |

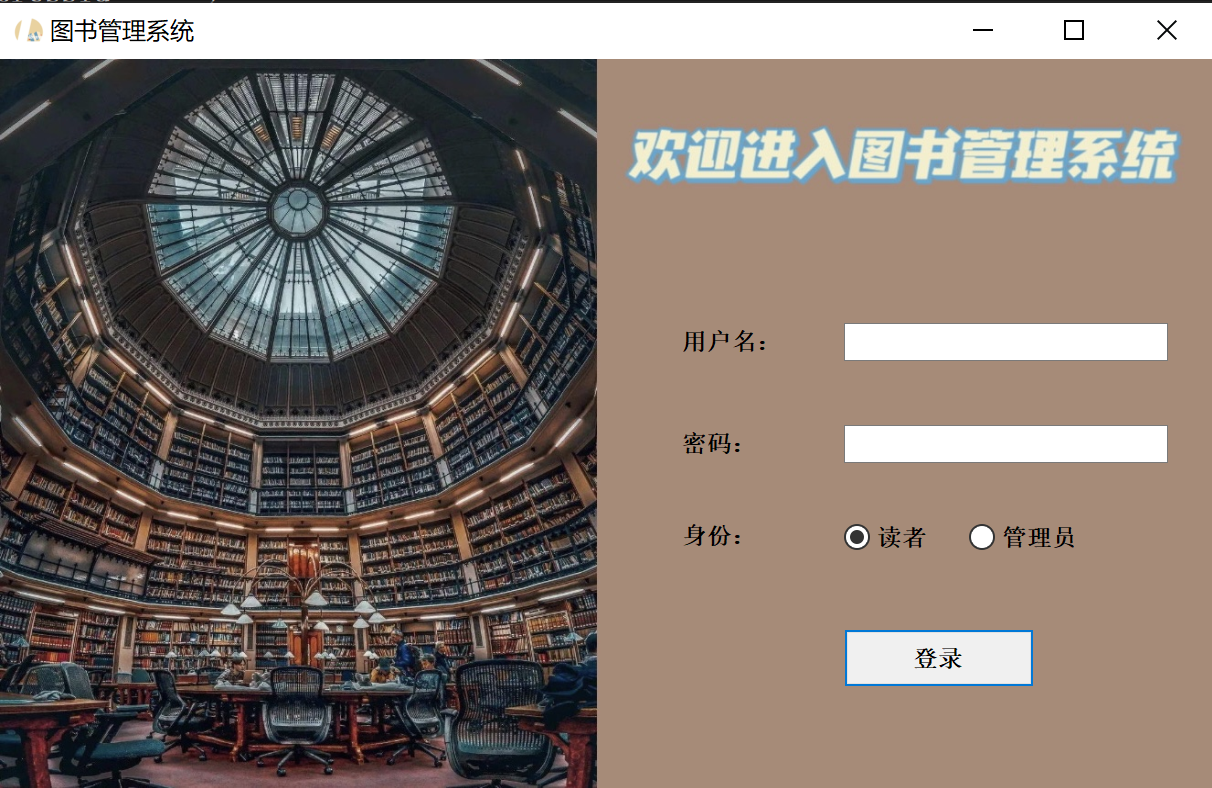
（9）个人违规记录表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| LeID | Int |  | 是 |  | 否 | 违规记录编号 |
| MID | Int |  |  | 是 | 否 | 管理员工号 |
| BorID | int |  |  | 是 | 否 | 借阅记录编号 |
| LeFine | int |  |  |  |  | 罚款金额 |
| LeReason | int |  |  |  |  | 违规原因（0为损坏，1为丢失） |
| LeTime | datetime |  |  |  |  | 操作时间 |
| LePay | int |  |  |  |  | 是否交款（0否，1是） |
| LeRemark | text |  |  |  |  | 备注 |

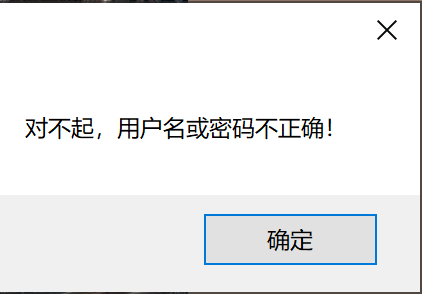
# 功能实现

## 1 登录

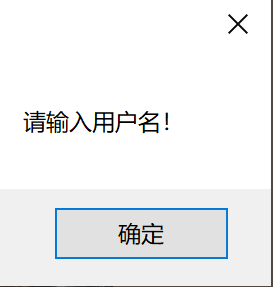
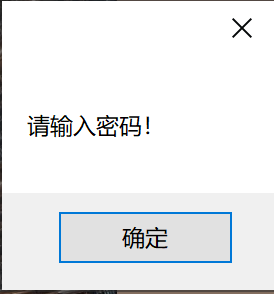
登录注册界面也即系统首页，用户输入用户名和密码，并选择身份后即可点击登录。



①若该用户不存在，则系统会弹出提示如下：



②若用户输入不完整（未输入用户名或密码），则系统会弹出提示如下：



该部分代码实现如下：

|  |
| --- |
| private void button1\_Click(object sender, EventArgs e)  {  if(username\_txt.Text.Trim()=="")  {  MessageBox.Show("请输入用户名！");  username\_txt.Focus();  }  else if (password\_txt.Text.Trim() == "")  {  MessageBox.Show("请输入密码！");  password\_txt.Focus();  }  else  {  string str;  if(rbt\_reader.Checked)  str = string.Format("select \* from reader\_info, rkind\_info where reader\_info.KID=rkind\_info.KID and RName='{0}' and Password='{1}'",username\_txt.Text,password\_txt.Text);  else  str = string.Format("select \* from manager\_info where MName='{0}' and Password='{1}'", username\_txt.Text, password\_txt.Text);  MySqlConnection conn = new MySqlConnection("server=localhost;database=library;UID=mysql;PWD=123456;port=3306");  conn.Open();  MySqlDataAdapter da = new MySqlDataAdapter(str,conn);  DataTable dt = new DataTable();  da.Fill(dt);  conn.Close();    if(dt.Rows.Count>0)  {  this.Hide();  if (rbt\_reader.Checked)  {  DataRow dr = dt.Rows[0];  int userid = Convert.ToInt16(dr["RID"].ToString());  int borTLimit = Convert.ToInt16(dr["TLimit"].ToString());  int borNLimit = Convert.ToInt16(dr["BLimit"].ToString()) ;  int readerBorNow = Convert.ToInt16(dr["RBorNow"].ToString());  Frm\_Main f = new Frm\_Main();  f.username = username\_txt.Text;  f.userid = userid;  f.borNLimit = borNLimit;  f.borTLimit= borTLimit;  f.borNow = readerBorNow;  f.Show();  }  else  {  ManMain f = new ManMain();  f.manname = username\_txt.Text;  f.Show();  }  }  else  {  MessageBox.Show("对不起，用户名或密码不正确！");  }  }  } |

## 2 读者主页

读者选择读者身份进入系统进入的界面如下图所示。该界面包含用户名（显示在界面上方）和用户可选择的功能。点击对应图标即可跳转进入对应功能界面。



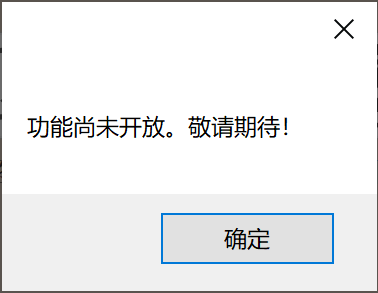
**①选择“借阅图书”功能**

点击对应按钮后，若系统记录**该用户的现借书籍数量已经达到借阅书籍上限**，则系统会提示用户现归还部分书籍后才能继续借阅。

|  |
| --- |
| private void btn\_Borrow\_Click(object sender, EventArgs e)  {  if(borNow>=borNLimit)  {  MessageBox.Show("您借阅的书籍本数已达上限，请先归还部分书籍后才能继续借阅哦！");  return;  }  reader\_BorrowFrm rb = new reader\_BorrowFrm();  rb.username = username;  rb.userid = userid;  rb.borTLimit = borTLimit;  rb.Show();  this.Hide();  } |

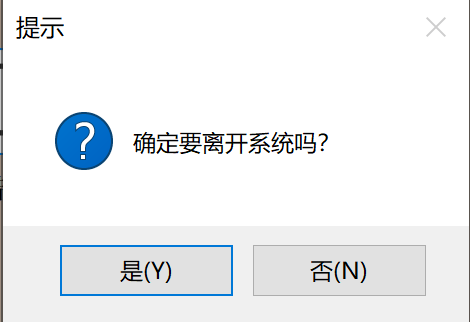
**②选择其余功能**

由于其余功能暂未开放，选择对应功能会弹出以下提示：



**③退出系统**

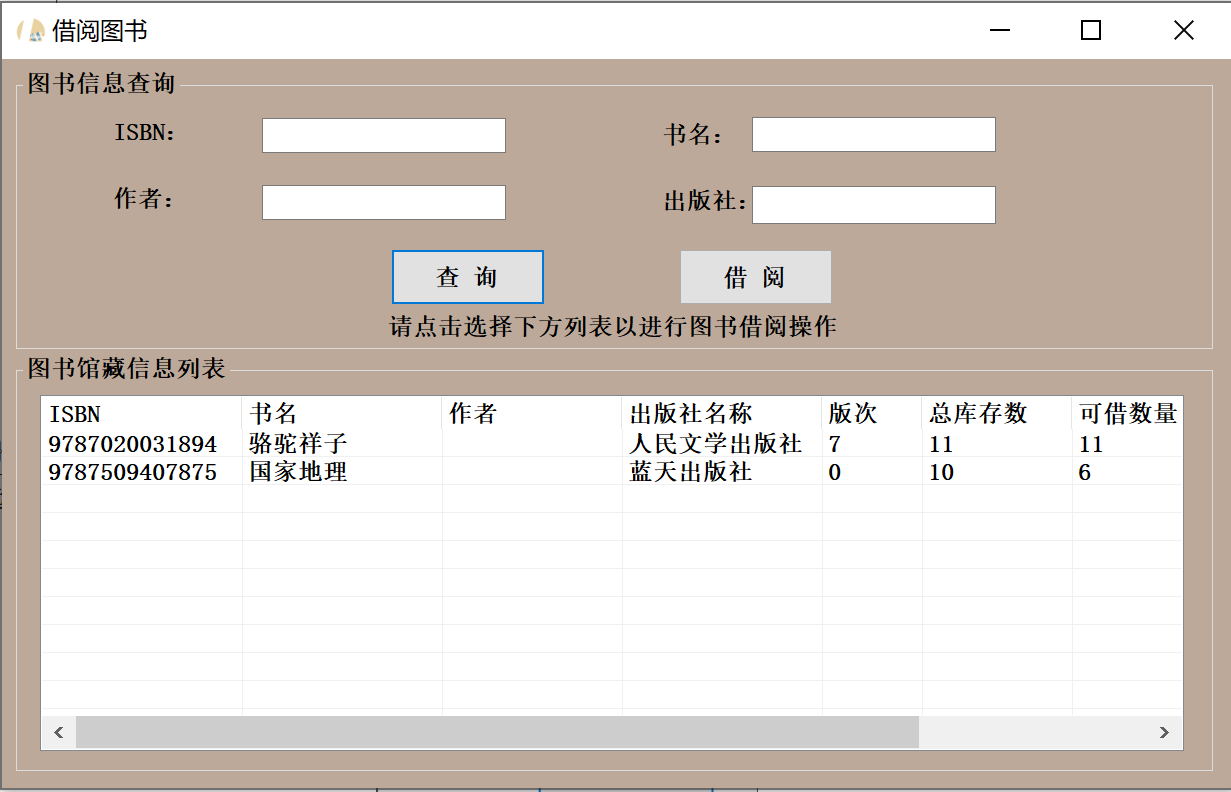
若读者在该界面点击右上角的关闭按钮，则会弹出以下提示：



若读者点击“是”，则会退出离开系统；若点击“否”则会返回读者主页。

## 3 借阅图书界面

在该界面，用户可输入ISBN、书名、作者、出版社进行**模糊查询（输入关键词即可）**，点击查询后列表更新。读者点击下方图书（不允许多选）后，点击“借阅”进行进一步的借阅操作。



**①“查询”功能**

该部分实现代码如下：

|  |
| --- |
| private void btn\_Search\_Click(object sender, EventArgs e)  {  string str = "select \* from book\_info, press\_info where book\_info.PID=press\_info.PID";  string str1="", str2="", str3 = "", str4 = "";  if (txt\_ISBN.Text!="")  {  string ISBN=txt\_ISBN.Text;  str1 = string.Format(" and ISBN='{0}'",ISBN);  }  if(txt\_BName.Text!="")  {  string BName=txt\_BName.Text;  str2 = string.Format(" and BName like '%{0}%'", BName);  }  if(txt\_Author.Text!="")  {  string author=txt\_Author.Text;  str3 = string.Format(" and BAuthor like '%{0}%'", author);  }  if(txt\_Press.Text!="")  {  string press = txt\_Press.Text;  str4 = string.Format(" and PName like '%{0}%'", press);  }  str=String.Concat(str,str1,str2,str3,str4);  get\_Book(str);  } |

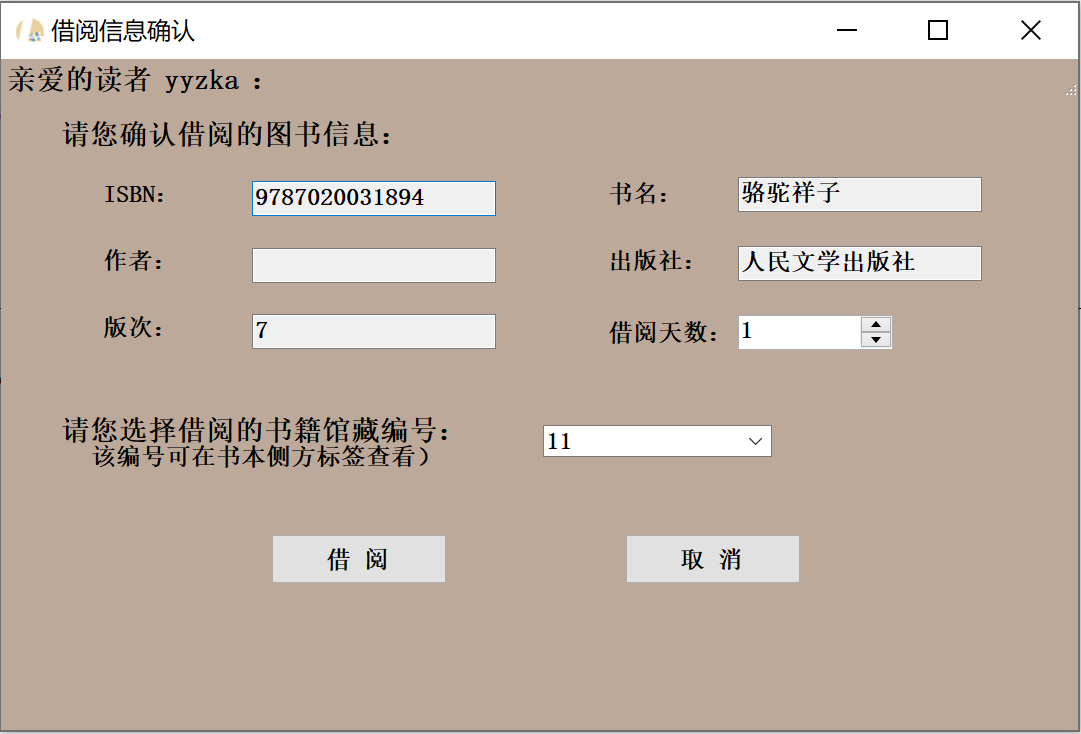
**②“借阅”部分功能**

若读者所选书籍无库存，则会弹出相应警告。

|  |
| --- |
| private void btn\_Borrow\_Click(object sender, EventArgs e)  {  if(lv\_Book.SelectedItems.Count>0)  {  ListViewItem item = lv\_Book.SelectedItems[0];  string ISBN = item.SubItems[0].Text;  string BName= item.SubItems[1].Text;  string author= item.SubItems[2].Text;  string press= item.SubItems[3].Text;  int version = Convert.ToInt16(item.SubItems[4].Text);  int stornow= Convert.ToInt16(item.SubItems[6].Text);  if(stornow==0)  {  MessageBox.Show("抱歉！您选择的书籍暂无库存！");  return;  }  else  {  reader\_Borrow2 r2=new reader\_Borrow2();  r2.ISBN = ISBN;  r2.author = author;  r2.press = press;  r2.BName=BName;  r2.version = version;  r2.username = username;  r2.userid = userid;  r2.borTLimit=borTLimit;  r2.Show();  this.Hide();  }  }  else  {  MessageBox.Show("请在列表中选择您要借阅的书籍！");  }  } |

## 4 借阅确认界面

在读者借阅界面点击“借阅”后，会进入该界面。借阅功能采用了**事务**进行实现。该界面会显示用户名、借阅的图书信息，用户需要自己输入**借阅天数（该天数上限会根据用户类别而异）**和借阅的书籍馆藏编号。



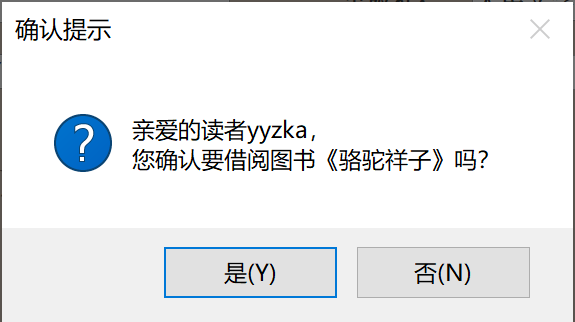
**①数据加载**

该部分代码如下：

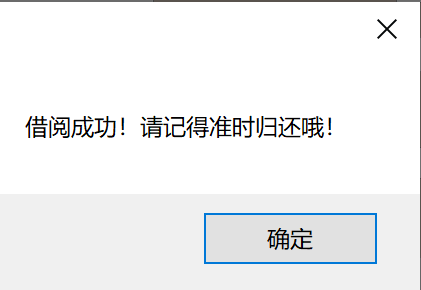
|  |
| --- |
| private void reader\_Borrow2\_Load(object sender, EventArgs e)  {  lbl\_username.Text = username;  databind\_BID();  txt\_Author.ReadOnly = false;  txt\_BName.ReadOnly = false;  txt\_ISBN.ReadOnly = false;  txt\_Press.ReadOnly = false;  txt\_Version.ReadOnly = false;  txt\_Author.Text = author;  txt\_Author.ReadOnly = true;  txt\_BName.Text = BName;  txt\_BName.ReadOnly = true;  txt\_ISBN.Text = ISBN;  txt\_ISBN.ReadOnly = true;  txt\_Press.Text = press;  txt\_Press.ReadOnly = true;  txt\_Version.Text=version.ToString();  txt\_Version.ReadOnly = true;  num\_Borlen.Maximum = borTLimit;  } |

**②“借阅”功能部分**

点击“借阅”后，会弹出“借阅确认”弹窗如下：



若用户点击“是”，则弹出借阅成功提示如下：



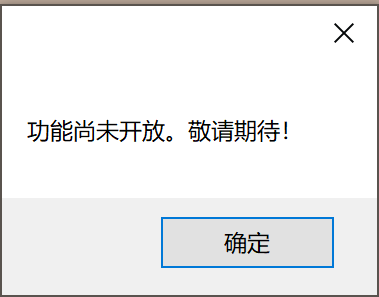
|  |
| --- |
| private void btn\_Borrow\_Click(object sender, EventArgs e)  {  int bookid = Convert.ToInt16(cbox\_BID.Text);  DialogResult dr = MessageBox.Show("亲爱的读者"+username+"，\n您确认要借阅图书《"+BName+"》吗？", "确认提示", MessageBoxButtons.YesNo, MessageBoxIcon.Question);  if(dr.Equals(DialogResult.Yes))  {  int borlen = Convert.ToInt16(num\_Borlen.Value.ToString());  MySqlConnection conn = new MySqlConnection ("server=localhost; database=library; UID=mysql; PWD=123456; port=3306");  conn.Open();  string str0 = "set @@autocommit=0;start transaction;";  string str1 = string.Format("insert into borrow\_info values(null,{0},1,{1},current\_date(),{2},0);", userid,bookid,borlen);  string str2 = string.Format("update inlib\_book\_info set LState=0 where LID={0};",bookid);  string str3 = string.Format("update book\_info set BBorrow=BBorrow+1, BStorNow=BStorNow-1 where ISBN='{0}';", ISBN);  string str4 = "commit;set @@autocommit=1;";  string str5 = string.Format("update reader\_info set RBor=RBor+1, RBorNow=RBorNow+1 where RID={0}", userid);  string str = String.Concat(str0, str1, str2, str3, str4,str5);  MySqlCommand cmd = new MySqlCommand(str, conn);  int i=cmd.ExecuteNonQuery();  if(i>0)  {  MessageBox.Show("借阅成功！请记得准时归还哦！");  }  else  {  MessageBox.Show("抱歉！借阅失败！请稍后重试！");  }  Frm\_Main f = new Frm\_Main();  f.userid = userid;  f.username = username;  f.Show();  this.Hide();  }  } |

## 5 管理员主页

管理员选择管理员身份进入系统进入的界面如下图所示。该界面包含用户名（显示在界面上方）和用户可选择的功能。点击对应图标即可跳转进入对应功能界面。



若用户选择“图书管理”，则会跳转至图书管理功能选择界面；若用户选择其余功能，则会弹出弹窗如下：



该部分实现代码如下：

|  |
| --- |
| private void btn\_bookmanagement\_Click(object sender, EventArgs e)  {  bookmanage\_frm bf = new bookmanage\_frm();  bf.username=manname;  bf.Show();  closeflag = false;  this.Close();  closeflag=true;  }  private void ManMain\_Load(object sender, EventArgs e)  {  lbl\_managername.Text = manname;  }  private void btn\_borrow\_Click(object sender, EventArgs e)  {  MessageBox.Show("功能尚未开放。敬请期待！");  }  private void btn\_returnmanagement\_Click(object sender, EventArgs e)  {  MessageBox.Show("功能尚未开放。敬请期待！");  }  private void btn\_fine\_Click(object sender, EventArgs e)  {  MessageBox.Show("功能尚未开放。敬请期待！");  } |

## 6 图书管理模块

该界面会显示管理员用户名，在该界面点击“出版社信息管理”或“图书基本信息管理”即可跳转至相应界面（“馆藏图书信息管理”暂未开放）。

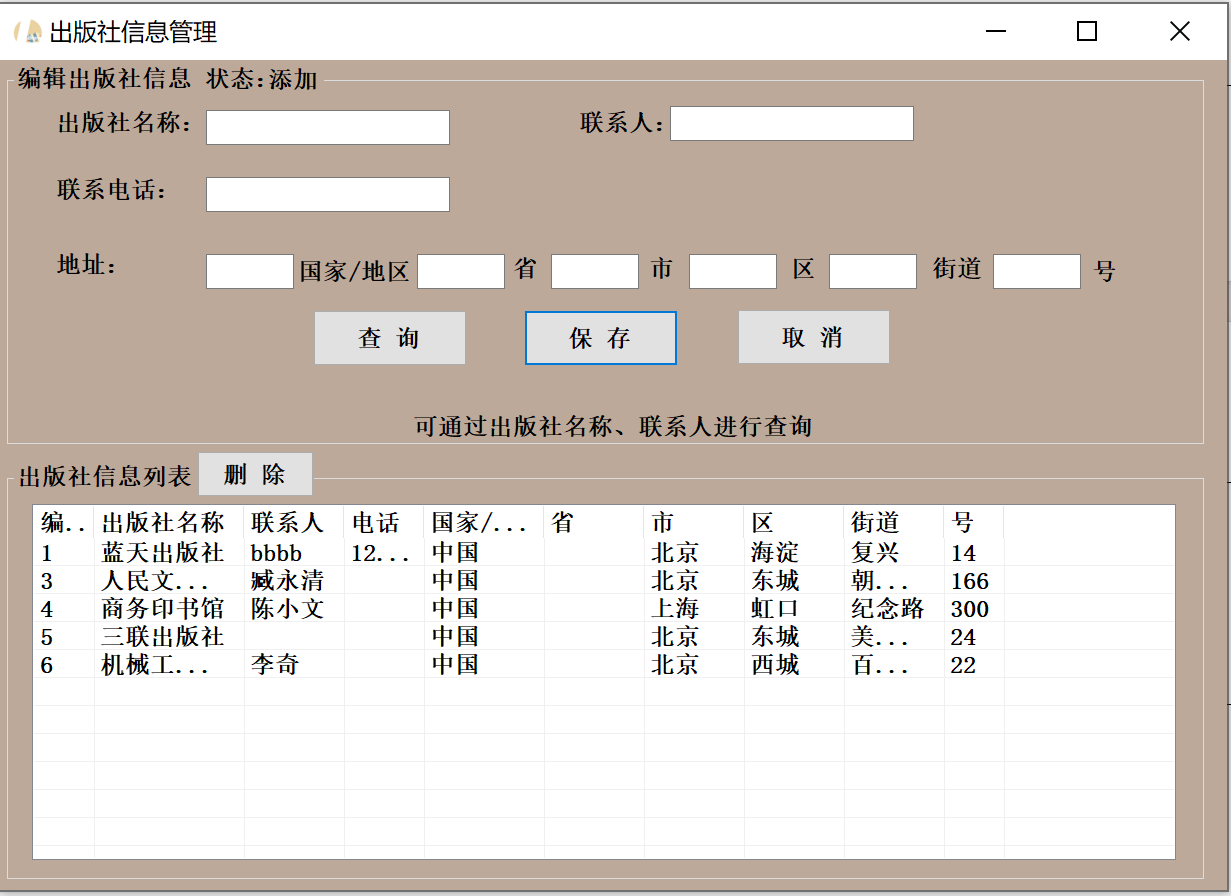


该部分实现代码如下：

|  |
| --- |
| private void btn\_Pressmanagement\_Click(object sender, EventArgs e)  {  Man\_Press mp = new Man\_Press();  mp.username = username;  mp.Show();  closeflag = true;  this.Close();  }  private void btn\_bookmanage\_Click(object sender, EventArgs e)  {  basicbookman\_frm bf=new basicbookman\_frm();  bf.username = username;  bf.Show();  closeflag = true;  this.Close();  } |

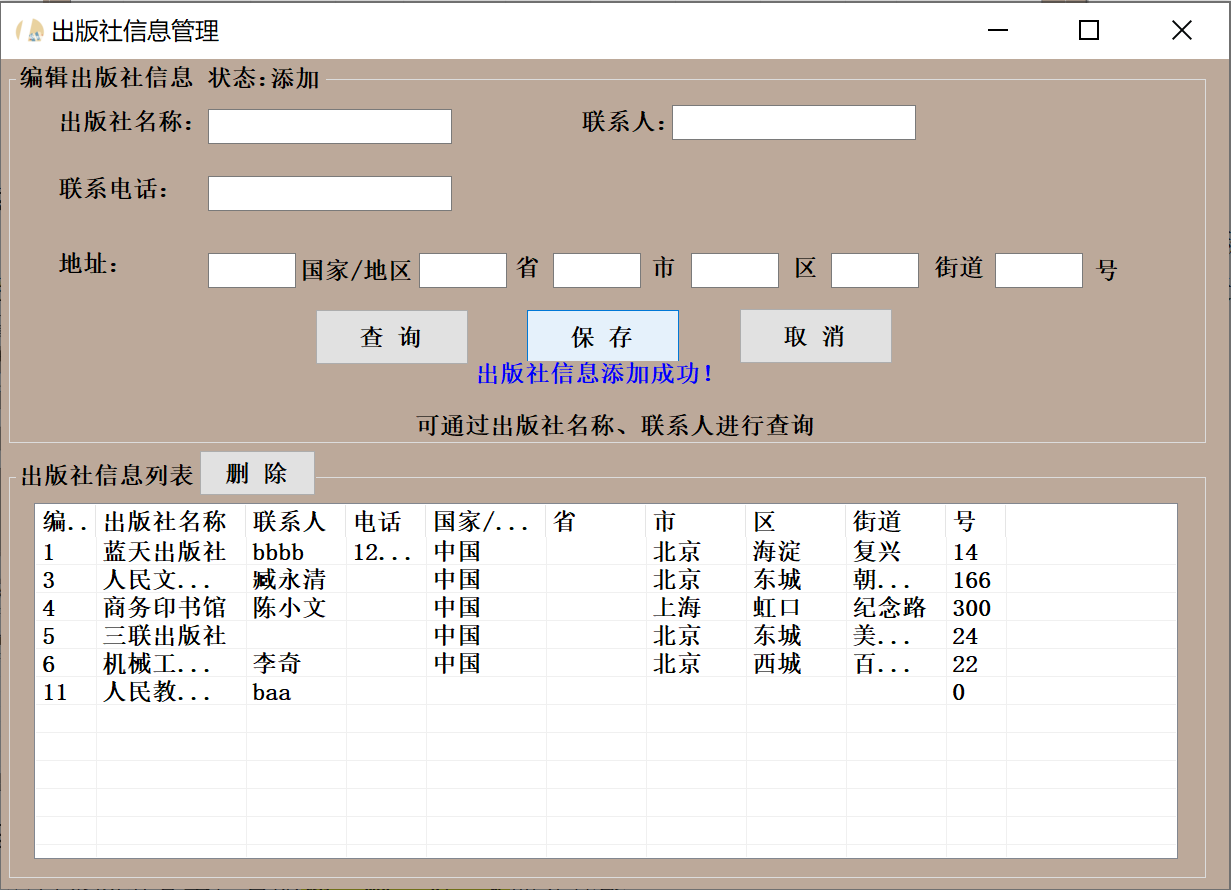
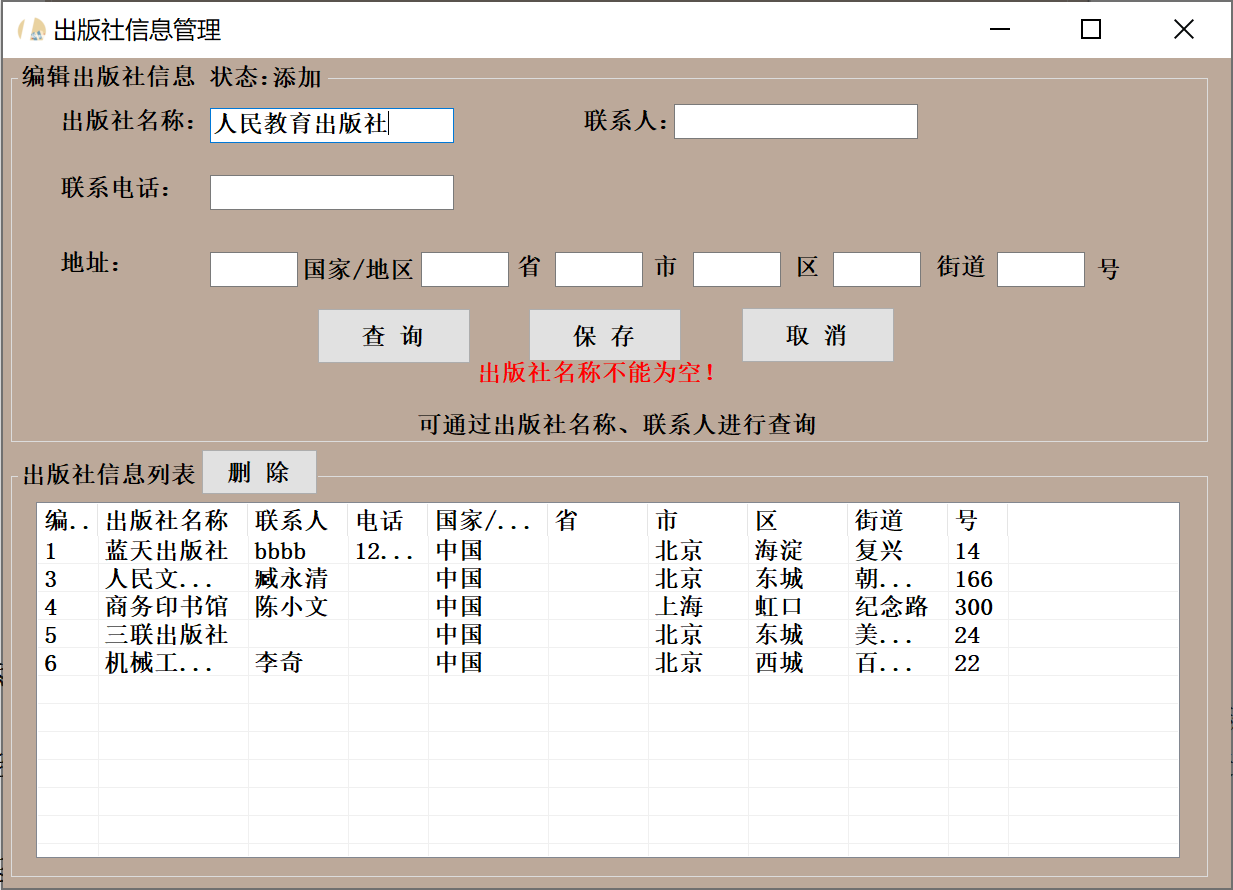
## 7 出版社信息管理

该界面包含出版社信息的**增、删、改、查**四个功能。



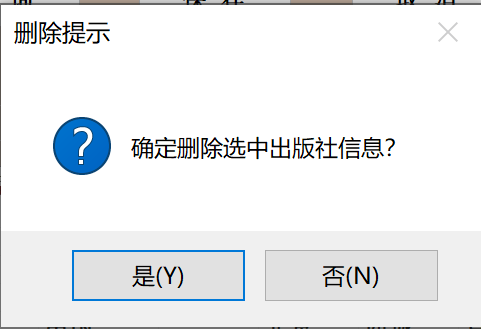
**①添加**

当界面上方的“状态”字段显示为“添加”时，用户可通过在上方输入出版社信息（出版社名称不能为空，否则会在下方显示红色提示，如左图所示）。当用户点击“保存”后，新书信息会即时更新在下方列表。



**②删除**

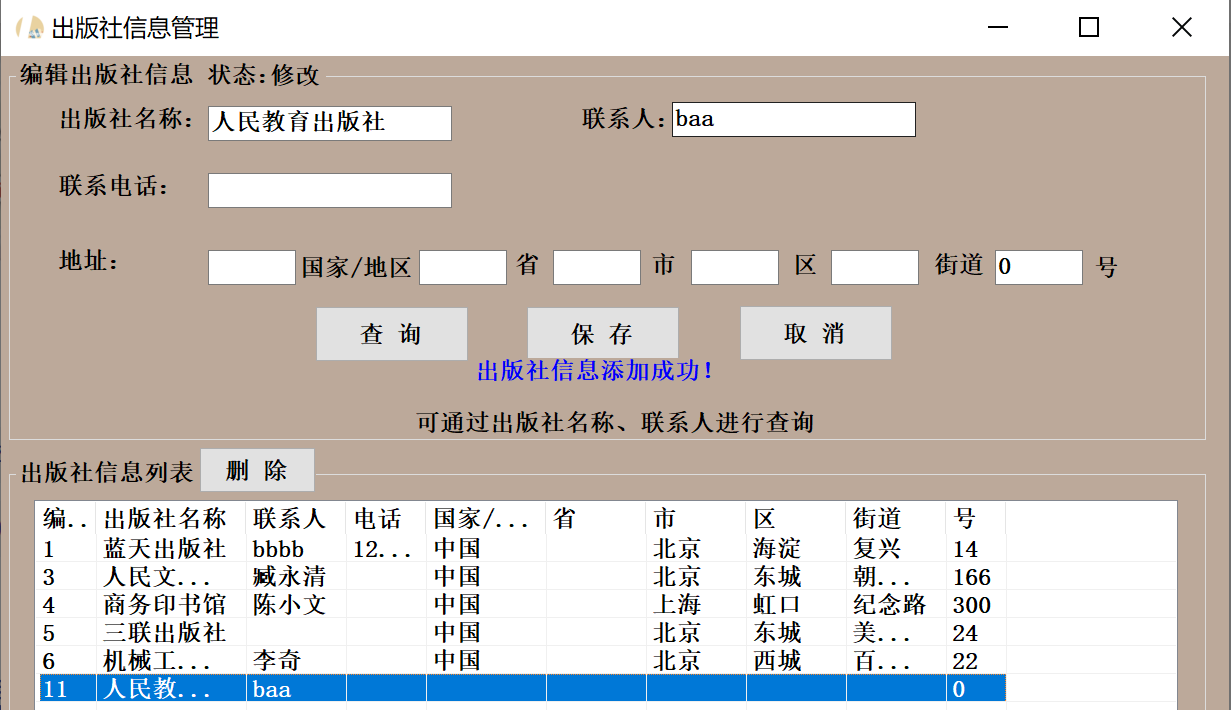
若用户选择下方列表中的出版社信息，并点击“删除”键，则系统会弹出如下提示：



用户点击“是”后即可删除该出版社信息，对应列表更新。

**③修改**

若用户选择下方列表的出版社信息，则界面上方“状态”字段变为“修改”，用户可直接在表格中进行相应修改，列表对应更新。



**④查询**

用户可通过出版社名称、联系人进行图书基本信息的**模糊查询**（即包含对应字段输入的词语即可查到）。



该部分实现代码如下：

|  |
| --- |
| //添加、修改  private void btn\_Save\_Click(object sender, EventArgs e)  {  string name = txt\_PName.Text.Trim();  string contact = txt\_ContactName.Text.Trim();  string tel = txt\_Telephone.Text.Trim();  string country = txt\_country.Text.Trim();  string pro = txt\_province.Text.Trim();  string city = txt\_city.Text.Trim();  string distinct = txt\_distinct.Text.Trim();  string street = txt\_street.Text.Trim();  string doorstr = txt\_doorno.Text.Trim();  int doorno=doorstr==""?0:Convert.ToInt16(doorstr);  if (name == "")  {  lbl\_Note.Text = "出版社名称不能为空！";  lbl\_Note.ForeColor = Color.Red;  txt\_PName.Focus();  }  else if (lbl\_Status.Text == "添加")  {  MySqlConnection conn = new MySqlConnection ("server=localhost; database=library; UID=mysql; PWD=123456; port=3306");  conn.Open();  string str = string.Format("insert into press\_info values(null,'{0}','{1}','{2}','{3}','{4}','{5}','{6}','{7}',{8})", name, contact, tel, country, pro,city,distinct,street,doorno);  MySqlCommand cmd = new MySqlCommand(str, conn);  int i = cmd.ExecuteNonQuery();  conn.Close();  if (i > 0)  {  lbl\_Note.Text = "出版社信息添加成功！";  lbl\_Note.ForeColor = Color.Blue;  clear\_Form();  get\_Press(";");  }  else  {  lbl\_Note.Text = "出版社信息添加失败！";  lbl\_Note.ForeColor = Color.Red;  }  }  else  {  MySqlConnection conn = new MySqlConnection ("server=localhost; database=library; UID=mysql; PWD=123456; port=3306");  conn.Open();  string str = string.Format("update press\_info set PName='{0}', PContact='{1}', PPhone='{2}', PCounRe='{3}', PProvince='{4}',PCity='{5}',PDistinct='{6}',PStreet='{7}',PDNo={8} where PID={9}", name, contact, tel, country, pro, city, distinct, street, doorno,pressid);  MySqlCommand cmd = new MySqlCommand(str, conn);  int i = cmd.ExecuteNonQuery();  conn.Close();  if (i > 0)  {  lbl\_Note.Text = "出版社信息修改成功！";  lbl\_Note.ForeColor = Color.Blue;  clear\_Form();  get\_Press(";");  }  else  {  lbl\_Note.Text = "出版社信息修改失败！";  lbl\_Note.ForeColor = Color.Red;  }  }  lv\_Customer.SelectedItems.Clear();  }  //删除  private void btn\_Del\_Click(object sender, EventArgs e)  {  if (pressid == "")  {  MessageBox.Show("请选择要删除的出版社信息！");  }  else  {  DialogResult dr = MessageBox.Show("确定删除选中出版社信息？", "删除提示", MessageBoxButtons.YesNo, MessageBoxIcon.Question);  if (dr == DialogResult.Yes)  {  MySqlConnection conn = new MySqlConnection ("server=localhost; database=library; UID=mysql; PWD=123456; port=3306");  conn.Open();  string str = string.Format("select \* from book\_info where PID={0}", pressid);  MySqlDataAdapter da = new MySqlDataAdapter(str, conn);  DataTable dt = new DataTable();  da.Fill(dt);  if (dt.Rows.Count>0)  {  MessageBox.Show("馆藏图书中有该出版社出版图书，该出版社信息不可删除！");  return;  }  str = string.Format("delete from press\_info where PID={0}", pressid);  MySqlCommand cmd = new MySqlCommand(str, conn);  int i = cmd.ExecuteNonQuery();  conn.Close();  if (i > 0)  {  lbl\_Note.Text = "出版社信息删除成功！";  lbl\_Note.ForeColor = Color.Blue;  get\_Press(";");  }  else  {  lbl\_Note.Text = "出版社信息删除失败！";  lbl\_Note.ForeColor = Color.Red;  }  }  pressid = "";  clear\_Form();  lv\_Customer.SelectedItems.Clear();  }  }  //查询  private void btn\_Search\_Click(object sender, EventArgs e)  {  string str1 = "", str2 = "";  if (txt\_PName.Text != "")  {  string pname = txt\_PName.Text;  str1 = string.Format(" PName='{0}'", pname);  }  if (txt\_ContactName.Text != "")  {  string cName = txt\_ContactName.Text;  str2 = string.Format("PContact like '%{0}%'", cName);  if (str1 != "")  str2 = String.Concat("and ", str2);  }  String str = String.Concat(" where ",str1, str2, ";");  get\_Press(str);  } |

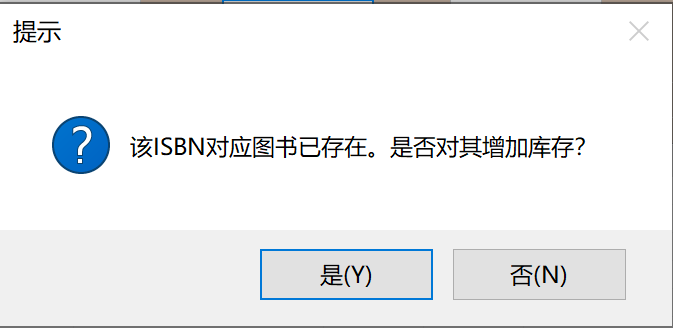
## 8 图书基本信息管理

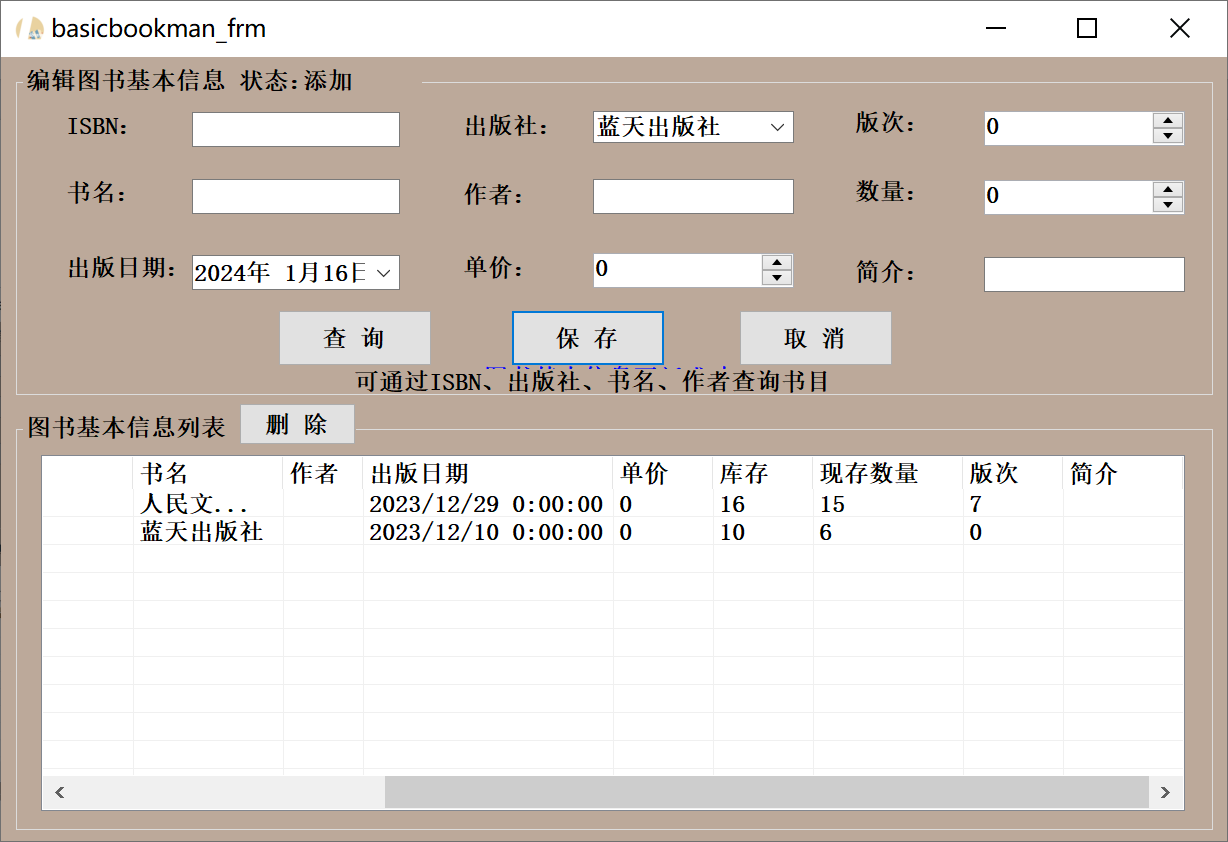
该界面包含图书基本信息的**增、删、改、查**四个功能。



**①添加**

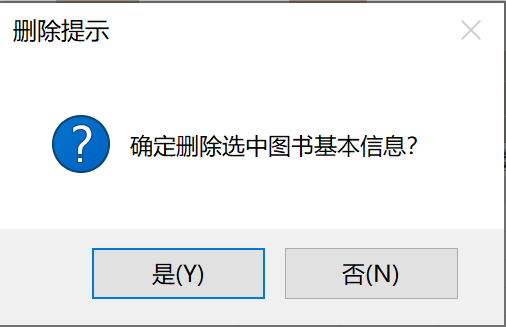
当界面上方的“状态”字段显示为“添加”时，用户可通过在上方输入书籍信息（ISBN、书名、出版社不能为空）。当用户点击“保存”后，新书信息会即时更新在下方列表。**若该书的ISBN和书名对应书籍已存在，则会弹出以下提示。**



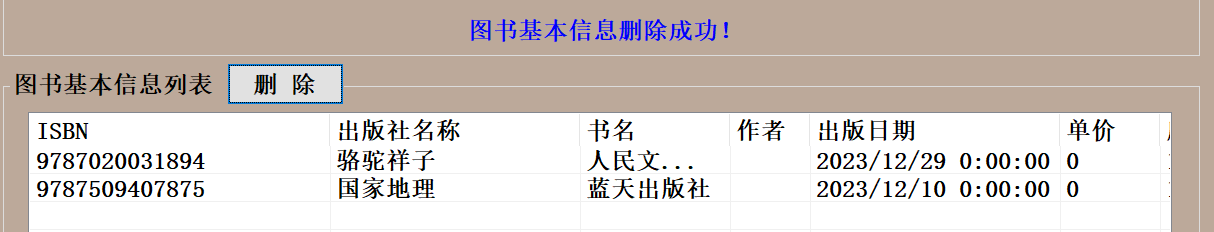
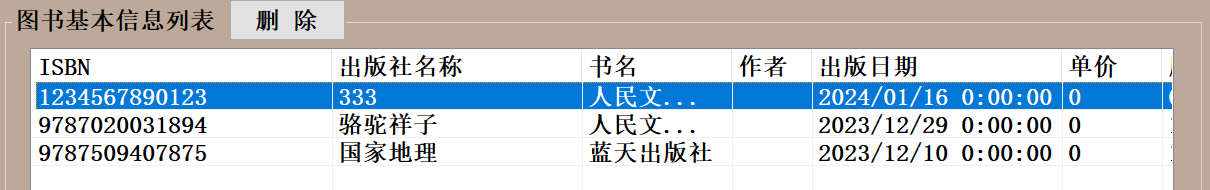


**②删除**

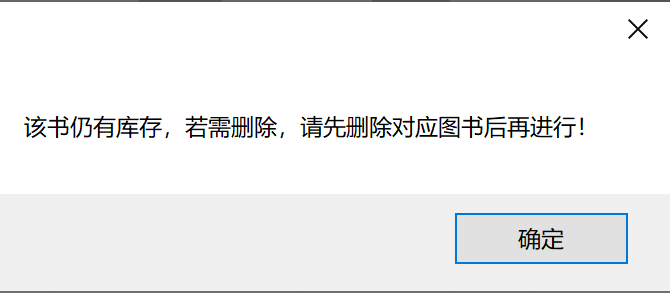
若用户选择下方列表中的图书基本信息，并点击“删除”键，则系统会弹出如下提示：



若用户点击“是”，则系统会判断该书是否已不存在于馆藏书籍中。若是，则会删除对应信息，列表相应发生更新。

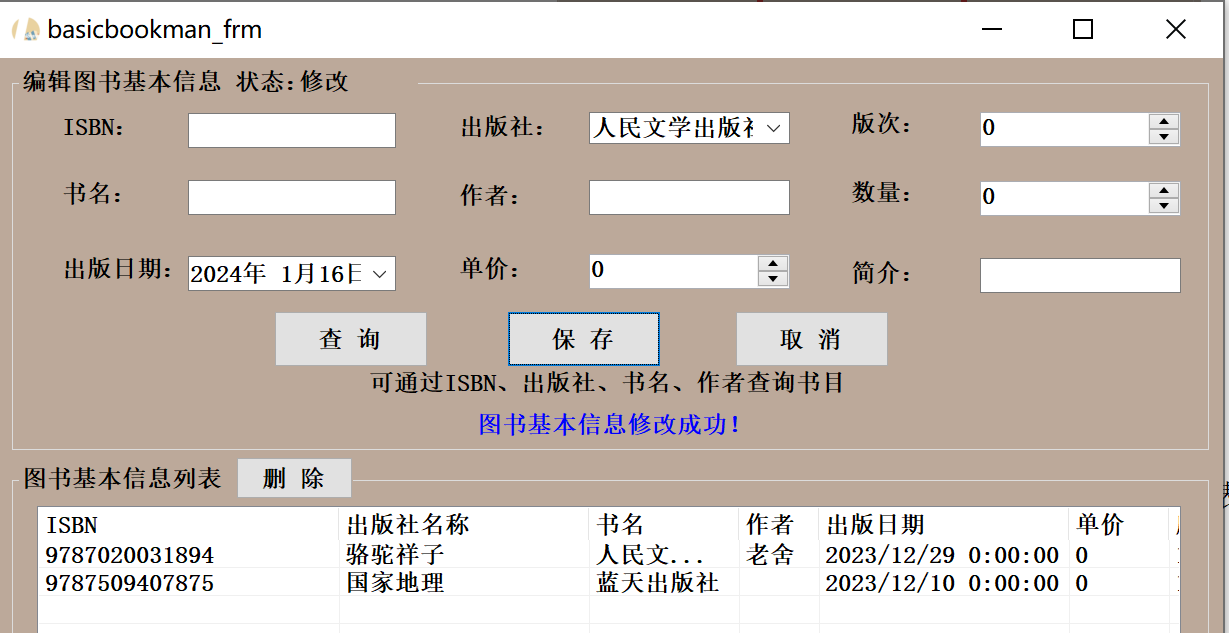
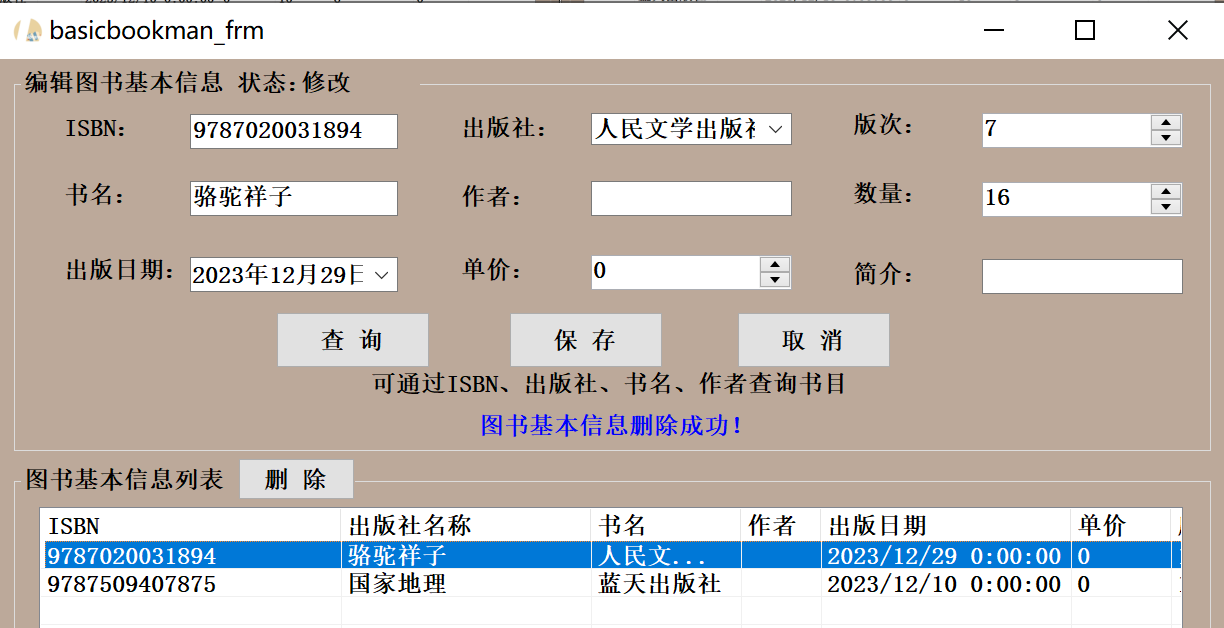


**若不是，则系统会弹出以下提示信息，删除操作中止。**



**③修改**

若用户选择下方列表的图书基本信息，则界面上方“状态”字段变为“修改”，用户可直接在表格中进行相应修改，列表对应更新。



**④查询**

用户可通过ISBN、出版社、书名、作者进行图书基本信息的**模糊查询**（即包含对应字段输入的词语即可查到）。



该部分代码实现如下：

|  |
| --- |
| //添加、修改  private void btn\_Save\_Click(object sender, EventArgs e)  {  string ISBN = txt\_ISBN.Text.Trim();  string bname = txt\_bookname.Text.Trim();  string author =txt\_author.Text.Trim();  string pubdate=dp\_pubtime.Value.ToString("yyyy-MM-dd");  string pricestr=num\_price.Value.ToString();  int price = Convert.ToInt16(pricestr);  string stostr=num\_storage.Value.ToString();  int storage = Convert.ToInt16(stostr);  string versionstr=num\_version.Value.ToString();  int version = Convert.ToInt16(versionstr);  string intro=txt\_intro.Text.Trim();  pressid = Convert.ToInt16(cbox\_press.SelectedValue.ToString());  if (ISBN == "")  {  lbl\_Note.Text = "ISBN不能为空！";  lbl\_Note.ForeColor = Color.Red;  txt\_ISBN.Focus();  }  else if(bname=="")  {  lbl\_Note.Text = "书名不能为空！";  lbl\_Note.ForeColor = Color.Red;  txt\_ISBN.Focus();  }  else if (lbl\_Status.Text == "添加")  {  MySqlConnection conn = new MySqlConnection ("server=localhost; database=library; UID=mysql; PWD=123456;port=3306");  conn.Open();  string hasISBN = string.Format("select COUNT(\*) from book\_info where ISBN='{0}'", ISBN);  MySqlCommand cmd = new MySqlCommand(hasISBN, conn);  int i = Convert.ToInt16(cmd.ExecuteScalar());  if (i > 0)  {  DialogResult dr = MessageBox.Show("该ISBN对应图书已存在。是否对其增加库存？", "提示", MessageBoxButtons.YesNo, MessageBoxIcon.Question);  if(dr == DialogResult.Yes)  {  string str1 = string.Format("update book\_info set BStorage=BStorage+{0}, BStorNow=BStorNow+{1} where ISBN='{2}'", storage,storage, ISBN);  cmd.CommandText = str1;  int i1 = cmd.ExecuteNonQuery();  string str2 = string.Format("insert into inlib\_book\_info values(null,'{0}',1,current\_date())",ISBN);  cmd.CommandText = str2;  int i2 = 0;  for(int j=0;j<storage;j++)  i2= cmd.ExecuteNonQuery();  if (i1>0&&i2>0)  {  lbl\_Note.Text = "图书基本信息更新成功！";  lbl\_Note.ForeColor = Color.Blue;  clear\_Form();  get\_basicBook(";");  }  else  {  lbl\_Note.Text = "图书基本信息更新失败！";  lbl\_Note.ForeColor = Color.Red;  }  }  else  {  clear\_Form();  }  }  else  {  if(ISBN.Length!=13)  {  MessageBox.Show("请输入长度为13位的ISBN编号！");  return;  }  string str = string.Format("insert into book\_info values('{0}',{1},'{2}','{3}','{4}',{5},{6},{7},{8},'{9}',null,null,null)", ISBN, pressid, bname, author, pubdate, price, storage, storage, version,intro);  cmd.CommandText = str;  int i1 = cmd.ExecuteNonQuery();    if (i1 > 0)  {  lbl\_Note.Text = "图书基本信息添加成功！";  lbl\_Note.ForeColor = Color.Blue;  clear\_Form();  get\_basicBook(";");  }  else  {  lbl\_Note.Text = "图书基本信息添加失败！";  lbl\_Note.ForeColor = Color.Red;  }  }  conn.Close();  }  else  {  MySqlConnection conn = new MySqlConnection("server=localhost;database=library;UID=mysql;PWD=123456;port=3306");  conn.Open();  string str = string.Format("update book\_info set BIntro='{0}',PID={1},BName='{2}',BAuthor='{3}',BPubDate='{4}',BPrice={5},BStorage={6},BStorNow={7},BVersion={8} where ISBN='{9}'",intro, pressid, bname, author, pubdate, price, storage, storage, version,ISBN);  MySqlCommand cmd = new MySqlCommand(str, conn);  int i = cmd.ExecuteNonQuery();  conn.Close();  if (i > 0)  {  lbl\_Note.Text = "图书基本信息修改成功！";  lbl\_Note.ForeColor = Color.Blue;  clear\_Form();  get\_basicBook(";");  }  else  {  lbl\_Note.Text = "图书基本信息修改失败！";  lbl\_Note.ForeColor = Color.Red;  }  }  lv\_basicBook.SelectedItems.Clear();  }  //删除  private void btn\_Del\_Click(object sender, EventArgs e)  {  if (theISBN == "")  {  MessageBox.Show("请选择要删除的图书基本信息！");  }  else  {  DialogResult dr = MessageBox.Show("确定删除选中图书基本信息？", "删除提示", MessageBoxButtons.YesNo, MessageBoxIcon.Question);  if (dr == DialogResult.Yes)  {  MySqlConnection conn = new MySqlConnection("server=localhost;database=library;UID=mysql;PWD=123456;port=3306");  conn.Open();  string selectstr = string.Format("select count(LID) from inlib\_book\_info where ISBN='{0}' group by ISBN; ", theISBN);  MySqlDataAdapter da = new MySqlDataAdapter(selectstr, conn);  DataTable dt = new DataTable();  da.Fill(dt);  if(dt.Rows.Count > 0)  {  MessageBox.Show("该书仍有库存，若需删除，请先删除对应图书后再进行！");  return;  }  string str = string.Format("delete from book\_info where ISBN='{0}'", theISBN);  MySqlCommand cmd = new MySqlCommand(str, conn);  int i = cmd.ExecuteNonQuery();  conn.Close();  if (i > 0)  {  lbl\_Note.Text = "图书基本信息删除成功！";  lbl\_Note.ForeColor = Color.Blue;  get\_basicBook(";");  }  else  {  lbl\_Note.Text = "图书基本信息删除失败！";  lbl\_Note.ForeColor = Color.Red;  }  }  theISBN = "";  clear\_Form();  lv\_basicBook.SelectedItems.Clear();  }  }  //查询  private void btn\_Search\_Click(object sender, EventArgs e)  {  string str1 = "", str2 = "", str3 = "", str4 = "";  if (txt\_ISBN.Text != "")  {  string ISBN = txt\_ISBN.Text;  str1 = string.Format(" and ISBN='{0}'", ISBN);  }  if (txt\_bookname.Text != "")  {  string BName = txt\_bookname.Text;  str2 = string.Format(" and BName like '%{0}%'", BName);  }  if (txt\_author.Text != "")  {  string author = txt\_author.Text;  str3 = string.Format(" and BAuthor like '%{0}%'", author);  }  if (cbox\_press.Text != "")  {  string press = cbox\_press.Text;  str4 = string.Format(" and PName like '%{0}%'", press);  }  String str = String.Concat(str1, str2, str3, str4,";");  get\_basicBook(str);  } |