网站技术文档

数据库内容：

一共五个数据表

一：tabLibrarian 管理员信息表

ID：每一个数据的标志，独一无二

数据类型：bigint

变化方式：按序加一

name: 存储管理员名的列

数据类型：nvarchar(20) 可输入中文，数字，字符 不允许空

pwd:存储密码的列

数据类型：nvarchar(20) 可输入中文，数字，字符 不允许空

二：tabBooks 图书信息表

ID：每一个数据的标志，独一无二

数据类型：bigint

变化方式：按序加一

bookname:存储书名的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

writer:存储作者的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

press:存储出版社的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

number:存储书籍编号的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

count:存储书籍数量的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

class1:存储书籍分类的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

三：tabUsers 用户信息表

ID：每一个数据的标志，独一无二

数据类型：int

变化方式：按序加一

name: 存储用户名的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

pwd:存储密码的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

sex:存储用户性别的列，只有男女两种

数据类型：nvarchar(50) 不允许空

phone:存储用户手机号的列

数据类型：bigint 不允许空

nameID:和用户手机号相同的列，用于建立数据库关系

数据类型：bigint 不允许空

question：用户密保问题存储区

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

answer:用户密保答案存储区

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

bookcount:存储用户借阅数目的列，初始值为0

数据类型：nvarchar(20) 不允许空

五：tabBorrow1 借阅信息表（为了形成数据关系图）

ID：每一个数据的标志，独一无二

数据类型：int

变化方式：按序加一

bookname:存储书名的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

writer:存储作者的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

press:存储出版社的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

number:存储书籍编号的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

count:存储书籍数量的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

class1:存储书籍分类的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

name: 存储用户名的列

数据类型：nvarchar(20) 可输入中文，数字，字符 不允许空

nameID:和用户手机号相同的列，用于建立数据库关系

数据类型：bigint 不允许空

四：tabBorrow 借阅信息表

ID：每一个数据的标志，独一无二

数据类型：int

变化方式：按序加一

bookname:存储书名的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

writer:存储作者的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

press:存储出版社的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

number:存储书籍编号的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

count:存储书籍数量的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

class1:存储书籍分类的列

数据类型：nvarchar(50) 可输入中文，数字，字符 不允许空

name: 存储用户名的列

数据类型：nvarchar(20) 可输入中文，数字，字符 不允许空

nameID:和用户手机号相同的列，用于建立数据库关系

数据类型：bigint 不允许空

类中函数

static string str = @"server=DESKTOP-VQ4TNA2;Integrated Security=SSPI;database=Library;";//数据库地址

public Class1()//类名

public static int Put(string sql)//打开数据库并执行SQL语句的函数，如执行成功返回1，否则返回0，主要用于数据库内容的增加和删除并得到操作结果；

{

try

{

SqlConnection conn = new SqlConnection(str);

DataTable ds = new DataTable();

conn.Open();

SqlDataAdapter da = new SqlDataAdapter(sql, conn);

da.Fill(ds);

conn.Close();

return 1;

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

return 0;

}

}

public static int Find(string sql)//用于查找数据库中是否含有符合条件的数据

{

try

{

SqlConnection conn = new SqlConnection(str);

DataTable dt = new DataTable();

conn.Open();

SqlDataAdapter da = new SqlDataAdapter(sql,conn);

da.Fill(dt);

int count = dt.Rows.Count;

conn.Close();

if (count> 0)

{

return 1;

}

else return 0;

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

return 0;

}

}

public static string Find1(string sql)//用于找回密码的函数

{

try

{

SqlConnection conn = new SqlConnection(str);

DataTable dt = new DataTable();

conn.Open();

SqlDataAdapter da = new SqlDataAdapter(sql,conn);

da.Fill(dt);

int count = dt.Rows.Count;

string name = dt.Rows.Count.ToString();

conn.Close();

string id=dt.Rows[0][0].ToString ();

return dt.Rows[0][2].ToString();

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

return "";

}

}

public static DataTable Put1(string sql)//用于数据绑定返回数据表的函数

{

SqlConnection conn = new SqlConnection(str);

DataTable ds = new DataTable();

conn.Open();

SqlDataAdapter da = new SqlDataAdapter(sql, conn);

da.Fill(ds);

conn.Close();

return ds;

}

public static bool bolNon(string temp)//用于判断输入是否是数字的函数

{

bool flag = true;

if (temp != "")

{

for (int i = 0; i < temp.Length; i++)

{

if (!Char.IsNumber(temp, i))

{ flag = false; break; }

}

}

else flag = false;

return flag;

}

public static string Find2(string sql)//返回查找到的日期

{

try

{

SqlConnection conn = new SqlConnection(str);

DataTable dt = new DataTable();

conn.Open();

SqlDataAdapter da = new SqlDataAdapter(sql, conn);

da.Fill(dt);

int count = dt.Rows.Count;

string name = dt.Rows.Count.ToString();

conn.Close();

string id = dt.Rows[0][0].ToString();

return dt.Rows[0][9].ToString();

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

return "";

}

}

数据库关系图

