

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC AN GIANG**

**KHOA CÔNG NGHỆ THÔNG TIN**





**ĐỒ ÁN**

**LẬP TRÌNH .NET**

**Chủ đề 03**

**XÂY DỰNG PHẦN MỀM**

**QUẢN LÝ SÁCH**

**Sinh viên thực hiện:**

Nguyễn Thị Yến Nhi - DTH185333

Nguyễn Hữu Lợi - DTH185311

Đào Thiện Nhơn - DTH185341

**GVHD: ThS Phạm Hữu Dũng**

An Giang, 29 tháng 11 năm 2020

**THÀNH VIÊN VÀ PHÂN CÔNG**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Họ tên** | **MSSV** | **Lớp** | **Nhiệm vụ** | **Tỉ lệ tham gia** |
| Nguyễn Hữu Lợi | DTH185311 | DH19TH1 | PowerPoint, Code | 100% |
| Nguyễn Thị Yến Nhi | DTH185331 | DH19TH1 | Form, Code, Word | 100% |
| Đào Thiện Nhơn | DTH185341 | DH19TH1 | Form, Code, Word | 100% |

MỤC LỤC

Phần 1 – PHÂN TÍCH TỔNG QUAN…………………………………………………4

1. Giới thiệu hệ thống…………………………………………………………………..4

2. Mục đích…………………………………………………………….……………….4

3. Yêu cầu……………………………………………………………...……………….5

4. Người dùng…………………………………………………………. ………………5

Phần 2 – PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG…………………………………..5

1. Phân tích bài toán…………………………………………………...……………….5

2. Xây dựng mô hình quan hệ………………………………………… ………………6

3. Thiết kế dữ liệu……………………………………………………..……………….6

Phần 3 – SƠ ĐỒ QUAN HỆ THỰC THỂ……………………………………………..8

Phần 4 – GIAO DIỆN CHƯƠNG TRÌNH…………………………………………….9

Phần 5 – CHƯƠNG TRÌNH LÀM ĐƯỢC NHỮNG GÌ? …………………………...36

Phần 6 – TÀI LIỆU THAM KHẢO………………………………………………….36

**QUẢN LÝ SÁCH**

**Phần 1 – PHÂN TÍCH TỔNG QUAN**

1. **GIỚI THIỆU HỆ THỐNG:**

* Ngày nay, cùng với sự bùng nổ và phát triển mạnh mẽ của Internet thì lĩnh vực Công nghệ thông tin đang dần trở thành một phần không thể thiếu trong cuộc sống lẫn công việc của con người. Nó đang ngày càng phát triển và ứng dụng mạnh mẽ trong các lĩnh vực quản lý sản xuất, kinh doanh, dịch vụ…Và lĩnh vực quản lý hàng hóa đã cho thấy hiệu quả của công nghệ khi áp dụng vào công tác quản lý, làm giảm thời gian và các công tác tính toán giấy tờ phức tạp, góp phần đáng kể trong việc thống kê tránh những sai sót trong kinh doanh.
* Đối với công việc Quản lý sách tương đối khó khăn và phức tạp bao gồm nhiều công việc với những nghiệp vụ khác nhau, cần nhiều thời gian và nhân lực, không tránh khỏi những sai sót nhầm lẫn. Cần có một giải pháp để đáp ứng và hỗ trợ tốt hơn cho công việc quản lý và kinh doanh của cửa hàng: đáp ứng các yêu cầu nghiệp vụ cơ bản, hỗ trợ truy cập tìm kiếm thông tin nhanh chóng và chính xác, giao diện trực quan, thân thiện, đưa ra các báo cáo chính xác cho nhà quản lý để lập kế hoạch kinh doanh trong thời gian tới. Từ những yêu cầu trên, việc xây dựng phần mềm “Quản lý sách” giúp giải quyết các vấn đề còn tồn tại, giảm thiểu chi phí và nâng cao kết quả kinh doanh.

1. **MỤC ĐÍCH:**

* Tự động hóa các thao tác thủ công trong việc quản lý sách, không phải quản lý thủ công (giấy tờ, sổ sách…)
* Một lượng lớn Sách gây khó khăn trong việc quản lý thủ công. Vì vậy phần mềm giúp công việc được xử lý nhanh hơn, bảo mật và chính xác hơn, giúp nhà quản lý nắm rõ thông tin bán hàng một cách chính xác.
* Tối ưu hóa công việc, giảm chi phí, góp phần nâng cao hiệu quả hoạt động cũng như dễ dàng nâng cấp và mở rộng hệ thống trong tương lai.

1. **YÊU CẦU:**

* Hệ thống đáp ứng được yêu cầu của nhà quản lý: quản lý truy xuất thông tin sách nhanh chóng, hỗ trợ tổng hợp, báo cáo, cập nhật khi có thay đổi.
* Các chức năng cơ bản: thêm, sửa, xóa thông tin sách.
* Giao diện trực quan, được bố trí hợp lí, dễ nhìn, dễ tương tác, tạo sự tiện lợi cho người quản lý.
* Định kỳ hàng tháng (quý/năm) thông tin về việc nhập/xuất sách sẽ được tổng hợp lại.
* Tính bảo mật và độ an toàn cao.
* Dễ dàng nâng cấp, bảo trì về sau.
* Cảnh báo tự động hàng tồn kho, sắp hết hàng.
* Phần mềm cho phép lưu trữ một lượng lớn thông tin.

1. **NGƯỜI DÙNG:**

* Người quản lý có thể sử dụng các chức năng của phần mềm Quản lý sách.
* Lập trình viên, bảo trì viên có chuyên môn sửa chữa và bảo trì hệ thống.

**Phần 2 – PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

1. **PHÂN TÍCH BÀI TOÁN:**

Yêu cầu xây dựng một hệ thống Quản lý sách của cửa hàng theo từng tháng, quý và năm. Cửa hàng quản lý các loại sách thông qua thể loại, nhà xuất bản, tác giả và mã sách. Mã sách là thông tin duy nhất để phân biệt các cuốn sách với nhau. Ngoài ra, hệ thống Quản lý sách còn quản lý thêm các thông tin khác:

* ***Quản lý thông tin Sách:***
* Thông tin về các loại sách: mã sách, tên sách, giá nhập, giá bán, số lượng tồn, mã thể loại, mã tác giả, mã NXB, năm xuất bản.
* Các chức năng: thêm, sửa, xóa, cập nhật.
* Hiển thị danh sách thông tin các loại sách.
  + ***Quản lý Nhà xuất bản:***
* Thông tin về NXB: mã NXB, tên NXB, số điện thoại, e – mail, địa chỉ.
* Các chức năng: thêm, sửa, xóa, cập nhật.
* Hiển thị danh sách thông tin các nhà xuất bản,
  + ***Quản lý Thể loại:***
* Thông tin về thể loại: mã thể loại, tên thể loại.
* Các chức năng: thêm, sửa, xóa, cập nhật.
* Hiển thị danh sách thông tin các thể loại.
  + ***Quản lý Tác giả:***
* Thông tin về tác giả: mã tác giả, tên tác giả, số điện thoại, e – mail, địa chỉ.
* Các chức năng: thêm, sửa, xóa, cập nhật.
* Hiển thị danh sách thông tin các tác giả.
  + ***Quản lý Bán hàng:***
* Thông tin về bán hàng: mã hóa đơn, mã sách, thời gian bán, số lượng bán, thành tiền.
* Các chức năng: thêm, sửa, xóa, cập nhật.
* Hiển thị danh sách và thông tin các quyển sách được bán.

Doanh thu sẽ được xử lý theo công thức nhất định, sau đó tổng hợp lại và được người quản lý xem xét.

Nếu trong quá trình xử lý doanh thu có sai sót thì thông qua hệ thống, người quản lý sẽ sứt dụng chức năng cập nhật để sửa chữa lại.

1. **XÂY DỰNG MÔ HÌNH QUAN HỆ:**

**SÁCH** (**Mã sách**, Tên sách, Mã thể loại, Giá nhập, Giá bán, Số lượng tồn, Mã tác giả, Mã NXB, Năm XB).

*(Mỗi cuốn Sách phải có một Mã sách để phân biệt Tên sách, Giá nhập, Giá bán, Số lượng tồn, Mã thể loại, Mã nhà xuất bản, Mã tác giả và Năm xuất bản)*

**NXB** (**Mã NXB**, Tên NXB, Số điện thoại, E – mail, Địa chỉ).

*(Mỗi NXB phải có một Mã NXB để phân biệt Tên NXB, Số điện thoại, E – mail, Địa chỉ NXB)*

**THỂ LOẠI** (**Mã thể loại**, Tên thể loại).

*(Mỗi Thể loại phải có Mã thể loại để phân biệt Tên thể loại)*

**TÁC GIẢ** (**Mã tác giả**, Tên tác giả, Số điện thoại, E – mail, Địa chỉ).

*(Mỗi Tác giả phải có một Mã tác giả để phân biệt Tên tác giả, Số điện thoại, E – mail và Địa chỉ)*

**BÁN HÀNG** (**Mã hóa đơn**, Mã sách, Thời gian bán, Số lượng bán, Thành tiền).

*(Mỗi Hóa đơn bán hàng phải có Mã hóa đơn, Mã sách, Thời gian bán để phân biệt Số lượng bán, Số lượng tồn và Giá bán)*

1. **THIẾT KẾ DỮ LIỆU:**

* **BẢNG SÁCH**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Ghi chú** |
| MaSach | Nvarchar (25) | PRIMARY KEY | Mã sách |
| TenSach | Nvarchar (250) |  | Tên sách |
| GiaNhap | Float |  | Giá nhập |
| GiaBan | Float |  | Giá bán |
| SLTon | Int |  | Số lượng tồn |
| MaTheLoai | Nvarchar (25) |  | Mã thể loại |
| MaTacGia | Nvarchar (25) |  | Mã tác giả |
| MaNXB | Nvarchar (25) |  | Mã NXB |
| NamXB | Int |  | Năm xuất bản |

* **BẢNG NXB**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Ghi chú** |
| MaNXB | Nvarchar (25) | PRIMARY KEY | Mã NXB |
| TenNXB | Nvarchar (250) |  | Tên NXB |
| SDT | Int |  | SĐT |
| E – mail | Nvarchar (250) |  | E – mail |
| DiaChi | Nvarchar (250) |  | Địa chỉ |

* **BẢNG THỂ LOẠI**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Ghi chú** |
| MaTheLoai | Nvarchar (25) | PRIMARY KEY | Mã thể loại |
| TenTheLoai | Nvarchar (250) |  | Tên thể loại |

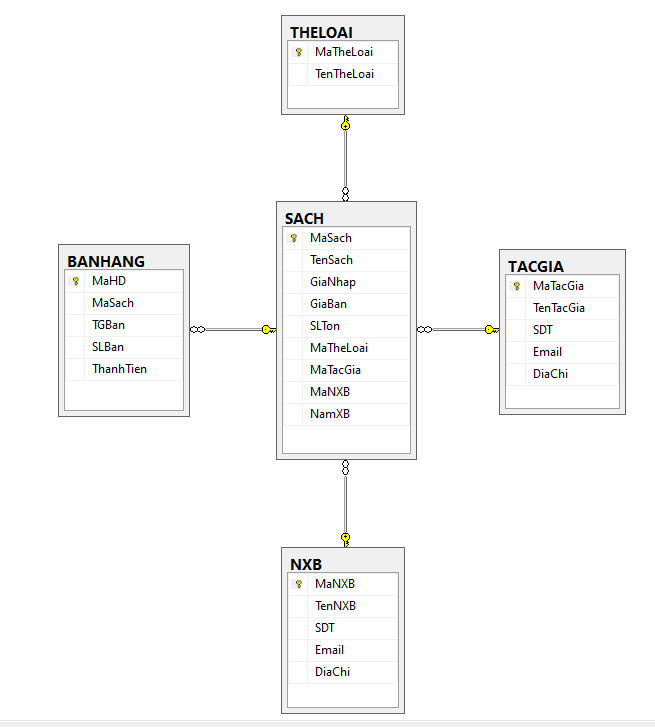
* **BẢNG TÁC GIẢ**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Ghi chú** |
| MaTacGia | Nvarchar (25) | PRIMARY KEY | Mã tác giả |
| TenTacGia | Nvarchar (250) |  | Tên tác giả |
| SDT | Int |  | SĐT |
| E – mail | Nvarchar (250) |  | E – mail |
| DiaChi | Nvarchar (250) |  | Địa chỉ |

* **BẢNG BÁN HÀNG**

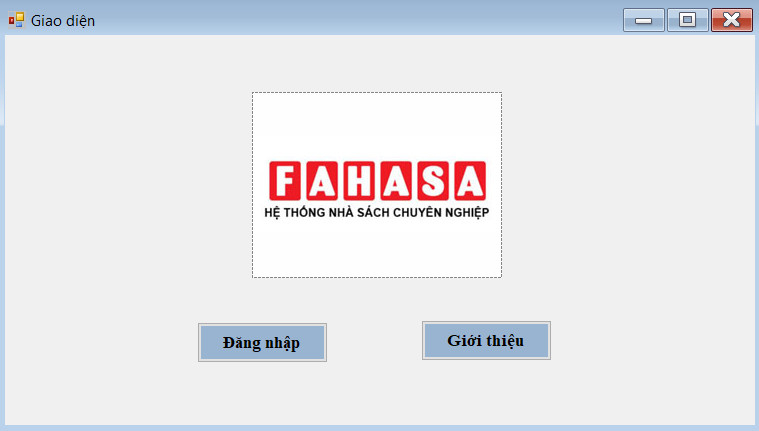
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Ghi chú** |
| MaHD | Nvarchar (25) | PRIMARY KEY | Mã hóa đơn |
| MaSach | Nvarchar (25) | PRIMARY KEY | Mã sách |
| TGBan | Datetime |  | Thời gian bán |
| SLBan | Int |  | Số lượng bán |
| ThanhTien | Float |  | Thành tiền |

**Phần 3 – SƠ ĐỒ QUAN HỆ THỰC THỂ**



**Phần 4 – GIAO DIỆN CHƯƠNG TRÌNH**

**FORM GIAO DIỆN**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace QLSach

{

public partial class frmGiaoDien : Form

{

public frmGiaoDien()

{

InitializeComponent();

}

private void btnDangNhap\_Click(object sender, EventArgs e)

{

frmDangNhap f = new frmDangNhap();

f.ShowDialog();

this.Hide();

}

private void btnGioiThieu\_Click(object sender, EventArgs e)

{

frmGioiThieu ff = new frmGioiThieu();

ff.ShowDialog();

this.Hide();

}

}

}

**FORM GIỚI THIỆU**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace QLSach

{

public partial class frmGioiThieu : Form

{

public frmGioiThieu()

{

InitializeComponent();

}

private void btnTroLai\_Click(object sender, EventArgs e)

{

frmGiaoDien f = new frmGiaoDien();

f.Show();

this.Hide();

}

private void btnDong\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Bạn có muốn thoát không?", "Thông báo",

MessageBoxButtons.OKCancel, MessageBoxIcon.Question) == DialogResult.OK)

{

Application.Exit();

}

}

}

}

**FORM ĐĂNG NHẬP**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

using QLSach;

namespace QLSach

{

public partial class frmDangNhap : Form

{

public frmDangNhap()

{

InitializeComponent();

}

private void btnDangNhap\_Click(object sender, EventArgs e)

{

if ((this.txtUser.Text == "") || (this.txtPass.Text == ""))

MessageBox.Show("Vui lòng nhập Tên người dùng hoặc Mật khẩu!", "Cảnh báo",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

if ((this.txtUser.Text == "nguyenvana") && (this.txtPass.Text == "123456"))

{

MessageBox.Show("Đăng nhập thành công!", "Thông báo");

frmTongQuan t = new frmTongQuan();

t.Show();

this.Hide(); //Ẩn form, các thuộc tính của nó vẫn tồn tại trong bộ nhớ

}

else

{

lblKQ.Text = "\*Tên đăng nhập hoặc Mật khẩu không đúng, hãy nhập lại!";

this.txtPass.Clear();

this.txtUser.Focus();

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

txtUser.Clear();

txtPass.Clear();

txtUser.Focus();

}

private void btnThoat\_Click(object sender, EventArgs e)

{

DialogResult traloi;

traloi = MessageBox.Show("Bạn có muốn đóng Hệ thống đăng nhập không?", "Xác nhận",

MessageBoxButtons.OKCancel, MessageBoxIcon.Question);

if (traloi == DialogResult.OK)

this.Close();

}

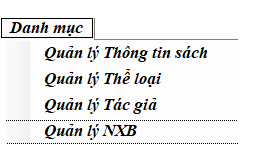
}

}

**FORM TỔNG QUAN**



* Trong phần Danh mục có các lựa chọn:

****

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace QLSach

{

public partial class frmTongQuan : Form

{

public frmTongQuan()

{

InitializeComponent();

}

private void quảnLýThôngTinSáchToolStripMenuItem\_Click(object sender, EventArgs e)

{

frmQLThongTinSach sach = new frmQLThongTinSach();

this.Hide(); //Ẩn form, các thuộc tính của nó vẫn tồn tại trong bộ nhớ

sach.ShowDialog();

this.Show(); //Để trở lại Form Tổng quan

}

private void quanLyTheLoaiToolStripMenuItem\_Click(object sender, EventArgs e)

{

frmTheLoai theloai = new frmTheLoai();

this.Hide(); //Ẩn form, các thuộc tính của nó vẫn tồn tại trong bộ nhớ

theloai.ShowDialog();

this.Show(); //Để trở lại Form Tổng quan

}

private void quảnLýTácGiảToolStripMenuItem\_Click(object sender, EventArgs e)

{

frmTacGia tacgia = new frmTacGia();

this.Hide(); //Ẩn form, các thuộc tính của nó vẫn tồn tại trong bộ nhớ

tacgia.ShowDialog();

this.Show(); //Để trở lại Form Tổng quan

}

private void quanLyNXBToolStripMenuItem\_Click(object sender, EventArgs e)

{

frmNXB nxb = new frmNXB();

this.Hide(); //Ẩn form, các thuộc tính của nó vẫn tồn tại trong bộ nhớ

nxb.ShowDialog();

this.Show(); //Để trở lại Form Tổng quan

}

private void banHangToolStripMenuItem\_Click(object sender, EventArgs e)

{

frmBanHang banhang = new frmBanHang();

this.Hide(); //Ẩn form, các thuộc tính của nó vẫn tồn tại trong bộ nhớ

banhang.ShowDialog();

this.Show(); //Để trở lại Form Tổng quan

}

private void dangXuatToolStripMenuItem1\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Bạn có muốn Đăng xuất khỏi hệ thống không?", "Thông báo",

MessageBoxButtons.OKCancel, MessageBoxIcon.Question) == DialogResult.OK)

{

frmGiaoDien giaodien = new frmGiaoDien();

this.Hide();

giaodien.ShowDialog();

this.Show();

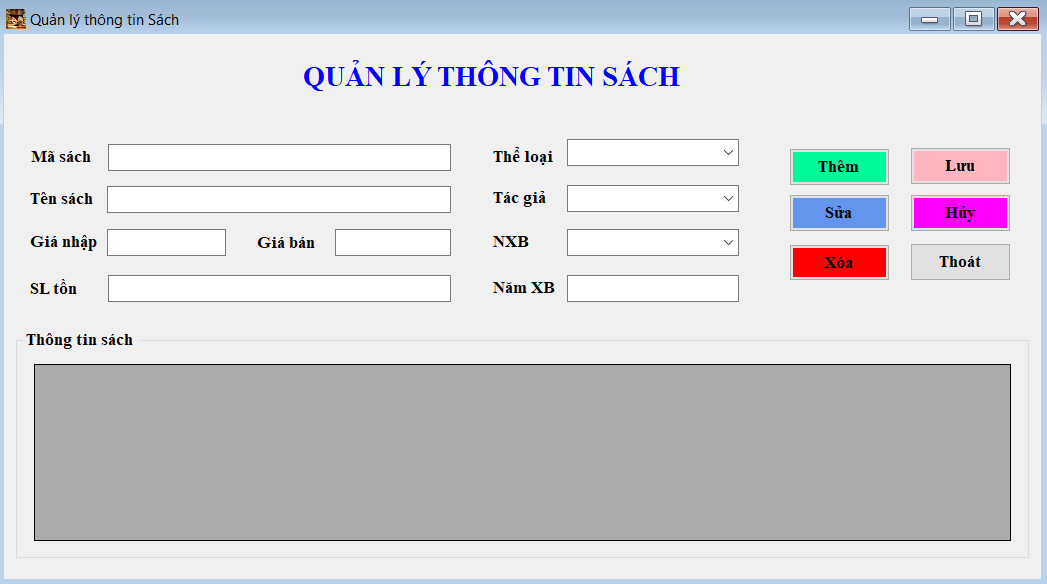
}

}

}

}

**FORM QUẢN LÝ THÔNG TIN SÁCH**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient; //Tạo chuỗi kết nối

namespace QLSach

{

public partial class frmQLThongTinSach : Form

{

public frmQLThongTinSach()

{

InitializeComponent();

}

SqlConnection conn = new SqlConnection();

//Khởi tại đối tượng DataSet

DataSet ds = new DataSet("dsQLSach");

SqlDataAdapter daSach;

SqlDataAdapter daNXB;

SqlDataAdapter daTheLoai;

SqlDataAdapter daTacGia;

private void frmQLThongTinSach\_Load(object sender, EventArgs e)

{

//Khai báo chuỗi kết nối để kết nối đến cơ sở dữ liệu QLSach

conn.ConnectionString = @"Data Source=LAPTOP-3SIUQHOQ\SQL;Initial Catalog=QLSach;Integrated Security=True";

//Dữ liệu combobox NXB

string sQueryNXB = @"select \* from NXB"; //Câu lệnh truy vấn để đưa vào DataAdapter

daNXB = new SqlDataAdapter(sQueryNXB, conn); //Tạo 1 đối tượng SqlDataAdapter tên daNXB

daNXB.Fill(ds, "tblNXB");

cmbNXB.DataSource = ds.Tables["tblNXB"]; //Đưa dữ liệu trong DataSet tên ds lên dataGirview

cmbNXB.DisplayMember = "TenNXB"; //Hiện tên trường

cmbNXB.ValueMember = "MaNXB"; //Lấy giá trị của trường

//Dữ liệu combobox Thể loại

string sQueryTheLoai = @"select \* from TheLoai";

daTheLoai = new SqlDataAdapter(sQueryTheLoai, conn);

daTheLoai.Fill(ds, "tblTheLoai"); //Lấy dữ liệu của table TheLoai đổ vào DataSet ds thông qua lệnh Select

cmbTheLoai.DataSource = ds.Tables["tblTheLoai"];

cmbTheLoai.DisplayMember = "TenTheLoai";

cmbTheLoai.ValueMember = "MaTheLoai";

//Dữ liệu combobox Tác giả

string sQueryTacGia = @"select \* from TacGia";

daTacGia = new SqlDataAdapter(sQueryTacGia, conn);

daTacGia.Fill(ds, "tblTacGia"); //Lấy dữ liệu của table TheLoai đổ vào DataSet ds thông qua lệnh Select

cmbTacGia.DataSource = ds.Tables["tblTacGia"];

cmbTacGia.DisplayMember = "TenTacGia";

cmbTacGia.ValueMember = "MaTacGia";

//Dữ liệu datagrid DS Sách

string sQuerySach = @"select n.\*, c.TenTheLoai, d.TenNXB, e.TenTacGia

from Sach n, TheLoai c, NXB d, TacGia e

where n.MaTheLoai = c.MaTheLoai and n.MaNXB = d.MaNXB and n.MaTacGia = e.MaTacGia";

daSach = new SqlDataAdapter(sQuerySach, conn);

daSach.Fill(ds, "tblSach");

dgSach.DataSource = ds.Tables["tblSach"];

//// … đặt tiêu đề tiếng Việt, định độ rộng cho các trường

dgSach.Columns["MaSach"].HeaderText = "Mã sách";

dgSach.Columns["MaSach"].Width = 90;

dgSach.Columns["TenSach"].HeaderText = "Tên sách";

dgSach.Columns["TenSach"].Width = 220;

dgSach.Columns["GiaNhap"].HeaderText = "Giá nhập";

dgSach.Columns["GiaNhap"].Width = 90;

dgSach.Columns["GiaBan"].HeaderText = "Giá bán";

dgSach.Columns["GiaBan"].Width = 80;

dgSach.Columns["SLTon"].HeaderText = "SL tồn";

dgSach.Columns["SLTon"].Width = 80;

dgSach.Columns["TenNXB"].HeaderText = "NXB";

dgSach.Columns["TenNXB"].Width = 80;

dgSach.Columns["NamXB"].HeaderText = "Năm XB";

dgSach.Columns["NamXB"].Width = 90;

dgSach.Columns["TenTheLoai"].HeaderText = "Thể loại";

dgSach.Columns["TenTheLoai"].Width = 120;

dgSach.Columns["TenTacGia"].HeaderText = "Tác giả";

dgSach.Columns["TenTacGia"].Width = 120;

dgSach.Columns["MaNXB"].Visible = false;

dgSach.Columns["MaTheLoai"].Visible = false;

dgSach.Columns["MaTacGia"].Visible = false;

//dgSach.Columns["MaSach"].Visible = false; //Ẩn row (cột) Mã Sách

//Tô màu xen kẽ cho Datagridview dgSach

dgSach.RowsDefaultCellStyle.BackColor = Color.White;

dgSach.AlternatingRowsDefaultCellStyle.BackColor = Color.LightCyan;

// Command Thêm Sách

string sThemSach = @"insert into Sach values(@MaSach, @TenSach, @GiaNhap, @GiaBan, @SLTon, @MaTheLoai, @MaTacGia, @MaNXB, @NamXB)";

SqlCommand cmThemSach = new SqlCommand(sThemSach, conn);

cmThemSach.Parameters.Add("@MaSach",SqlDbType.NVarChar, 10,"MaSach");

cmThemSach.Parameters.Add("@TenSach", SqlDbType.NVarChar, 50, "TenSach");

cmThemSach.Parameters.Add("@GiaNhap", SqlDbType.NVarChar, 10, "GiaNhap");

cmThemSach.Parameters.Add("@GiaBan", SqlDbType.NVarChar, 10, "GiaBan");

cmThemSach.Parameters.Add("@SLTon", SqlDbType.Int, 3, "SLTon");

cmThemSach.Parameters.Add("@MaTheLoai", SqlDbType.NVarChar, 50, "MaTheLoai");

cmThemSach.Parameters.Add("@MaTacGia", SqlDbType.NVarChar, 25, "MaTacGia");

cmThemSach.Parameters.Add("@MaNXB", SqlDbType.NVarChar, 25, "MaNXB");

cmThemSach.Parameters.Add("@NamXB", SqlDbType.NVarChar, 25, "NamXB");

daSach.InsertCommand = cmThemSach;

//Command Sửa Sách

string sSuaSach = @"update Sach set TenSach=@TenSach, GiaNhap=@GiaNhap, GiaBan=@GiaBan, SLTon=@SLTon,

MaTheLoai=@MaTheLoai, MaTacGia=@MaTacGia, MaNXB=@MaNXB, NamXB=@NamXB

where MaSach=@MaSach";

SqlCommand cmSuaSach = new SqlCommand(sSuaSach, conn);

cmSuaSach.Parameters.Add("@MaSach", SqlDbType.NVarChar, 10, "MaSach");

cmSuaSach.Parameters.Add("@TenSach", SqlDbType.NVarChar, 50, "TenSach");

cmSuaSach.Parameters.Add("@GiaNhap", SqlDbType.NVarChar, 10, "GiaNhap");

cmSuaSach.Parameters.Add("@GiaBan", SqlDbType.NVarChar, 10, "GiaBan");

cmSuaSach.Parameters.Add("@SLTon", SqlDbType.Int, 3, "SLTon");

cmSuaSach.Parameters.Add("@MaTheLoai", SqlDbType.NVarChar, 50, "MaTheLoai");

cmSuaSach.Parameters.Add("@MaTacGia", SqlDbType.NVarChar, 25, "MaTacGia");

cmSuaSach.Parameters.Add("@MaNXB", SqlDbType.NVarChar, 25, "MaNXB");

cmSuaSach.Parameters.Add("@NamXB", SqlDbType.NVarChar, 25, "NamXB");

daSach.UpdateCommand = cmSuaSach;

//Command Xóa Sách

string sXoaSach = @"delete from Sach where MaSach = @MaSach";

SqlCommand cmXoaSach = new SqlCommand(sXoaSach, conn);

cmXoaSach.Parameters.Add("@MaSach", SqlDbType.NVarChar, 5, "MaSach");

daSach.DeleteCommand = cmXoaSach;

}

private void dgSach\_Click(object sender, EventArgs e)

{

DataGridViewRow dr = dgSach.SelectedRows[0];

txtMaSach.Text = dr.Cells["MaSach"].Value.ToString();

txtTenSach.Text = dr.Cells["TenSach"].Value.ToString();

txtGiaNhap.Text = dr.Cells["GiaNhap"].Value.ToString();

txtGiaBan.Text = dr.Cells["GiaBan"].Value.ToString();

txtSLTon.Text = dr.Cells["SLTon"].Value.ToString();

cmbTheLoai.Text = dr.Cells["TenTheLoai"].Value.ToString();

cmbTacGia.Text = dr.Cells["TenTacGia"].Value.ToString();

cmbNXB.Text = dr.Cells["TenNXB"].Value.ToString();

txtNamXB.Text = dr.Cells["NamXB"].Value.ToString();

}

private void dgSach\_CellClick(object sender, DataGridViewCellEventArgs e)

{

//Khi chọn 1 dòng trên DataGridView thông tin sẽ hiển thị lên các control tương ứng trên Form

DataGridViewRow dr = dgSach.SelectedRows[0];

txtMaSach.Text = dr.Cells["MaSach"].Value.ToString();

txtTenSach.Text = dr.Cells["TenSach"].Value.ToString();

txtGiaNhap.Text = dr.Cells["GiaNhap"].Value.ToString();

txtGiaBan.Text = dr.Cells["GiaBan"].Value.ToString();

txtSLTon.Text = dr.Cells["SLTon"].Value.ToString();

cmbTheLoai.Text = dr.Cells["TenTheLoai"].Value.ToString();

cmbTacGia.Text = dr.Cells["TenTacGia"].Value.ToString();

cmbNXB.Text = dr.Cells["TenNXB"].Value.ToString();

txtNamXB.Text = dr.Cells["NamXB"].Value.ToString();

}

//Kiểm tra Mã sách trùng

public int kiemtra()

{

for (int i = 0; i < dgSach.RowCount - 1; i++)

{

if (txtMaSach.Text == dgSach.Rows[i].Cells["MaSach"].Value.ToString())

return 0;

}

return 1;

}

private void btnThem\_Click(object sender, EventArgs e)

{

if (txtMaSach.Text == "" || txtTenSach.Text == "" || txtGiaNhap.Text == "" || txtGiaBan.Text == "" || txtSLTon.Text == "" ||

cmbTheLoai.Text == "" || cmbTacGia.Text == "" || cmbNXB.Text == "" || txtNamXB.Text == "")

{

MessageBox.Show("Bạn chưa nhập đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

if (kiemtra() == 0)

MessageBox.Show("Đã tồn tại Mã sách!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

else

{

//Thêm 1 dòng vào bảng tblSach

DataRow row = ds.Tables["tblSach"].NewRow();

row["MaSach"] = txtMaSach.Text;

row["TenSach"] = txtTenSach.Text;

row["GiaNhap"] = txtGiaNhap.Text;

row["GiaBan"] = txtGiaBan.Text;

row["SLTon"] = txtSLTon.Text;

row["TenTheLoai"] = cmbTheLoai.Text;

row["MaTheLoai"] = cmbTheLoai.SelectedValue;

row["TenTacGia"] = cmbTacGia.Text;

row["MaTacGia"] = cmbTacGia.SelectedValue;

row["TenNXB"] = cmbNXB.Text;

row["MaNXB"] = cmbNXB.SelectedValue;

row["NamXB"] = txtNamXB.Text;

ds.Tables["tblSach"].Rows.Add(row); //Đưa các dòng này vào table tblSach

}

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

if (txtTenSach.Text == "" || txtGiaNhap.Text == "" || txtGiaBan.Text == "" || txtSLTon.Text == "" ||

cmbTheLoai.Text == "" || cmbTacGia.Text == "" || cmbNXB.Text == "" || txtNamXB.Text == "")

{

MessageBox.Show("Bạn chưa nhập đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

DataGridViewRow dr = dgSach.SelectedRows[0];

dgSach.BeginEdit(true);

dr.Cells["MaSach"].Value = txtMaSach.Text;

dr.Cells["TenSach"].Value = txtTenSach.Text;

dr.Cells["GiaNhap"].Value = txtGiaNhap.Text;

dr.Cells["GiaBan"].Value = txtGiaBan.Text;

dr.Cells["SLTon"].Value = txtSLTon.Text;

dr.Cells["TenTheLoai"].Value = cmbTheLoai.Text;

dr.Cells["MaTheLoai"].Value = cmbTheLoai.SelectedValue;

dr.Cells["TenTacGia"].Value = cmbTacGia.Text;

dr.Cells["MaTacGia"].Value = cmbTacGia.SelectedValue;

dr.Cells["TenNXB"].Value = cmbNXB.Text;

dr.Cells["MaNXB"].Value = cmbNXB.SelectedValue;

dr.Cells["NamXB"].Value = txtNamXB.Text;

dgSach.EndEdit();

MessageBox.Show("Đã cập nhật vào Thông tin Sách!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

if(dgSach.SelectedRows.Count > 0) //Lệnh đếm, dòng đã được chọn

{

dgSach.Rows.Remove(dgSach.SelectedRows[0]);

txtMaSach.Clear();

txtTenSach.Clear();

txtGiaNhap.Clear();

txtGiaBan.Clear();

txtSLTon.Clear();

cmbTheLoai.ResetText();

cmbTacGia.ResetText();

cmbNXB.ResetText();

txtNamXB.Clear();

txtMaSach.Focus();

}

}

private void btnLuu\_Click(object sender, EventArgs e)

{

this.daSach.Update(ds, "tblSach");

dgSach.Refresh();

MessageBox.Show("Lưu thành công!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void btnHuy\_Click(object sender, EventArgs e)

{

MessageBox.Show("Bạn có muốn hủy các thao tác vừa rồi không?", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Question);

{

ds.Tables["tblSach"].RejectChanges(); //Hủy các thao tác trước khi lưu

}

}

private void btnThoat\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Bạn có muốn thoát không?", "Thông báo",

MessageBoxButtons.OKCancel, MessageBoxIcon.Question) == DialogResult.OK)

{

Application.Exit();

}

}

private void txtGiaNhap\_TextChanged(object sender, EventArgs e)

{

double gianhap, giaban;

if(txtGiaNhap.Text == "")

{

gianhap = 0;

}

else

{

gianhap = Convert.ToDouble(txtGiaNhap.Text);

}

giaban = gianhap + (gianhap \* 0.1);

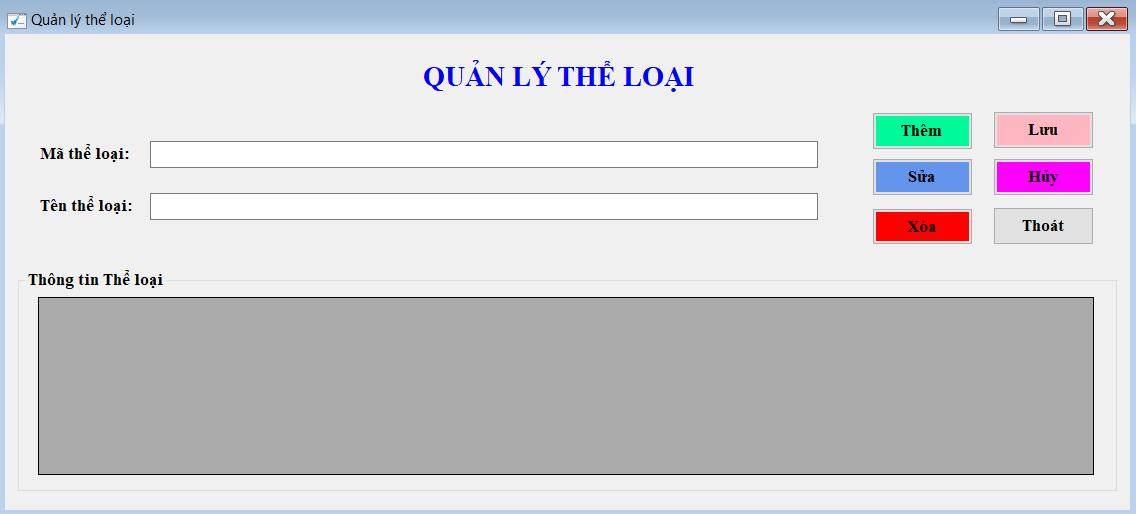
txtGiaBan.Text = giaban.ToString();

}

}

}

**FORM QUẢN LÝ THỂ LOẠI**

****

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace QLSach

{

public partial class frmTheLoai : Form

{

public frmTheLoai()

{

InitializeComponent();

}

SqlConnection conn = new SqlConnection();

//Khởi tại đối tượng DataSet

DataSet ds = new DataSet();

SqlDataAdapter daTheLoai;

private void frmTheLoai\_Load(object sender, EventArgs e)

{

conn.ConnectionString = @"Data Source=LAPTOP-3SIUQHOQ\SQL;Initial Catalog=QLSach;Integrated Security=True";

//Dữ liệu Datagridview NXB

string sQueryTheLoai = @"select \* from TheLoai"; //Câu lệnh truy vấn để đưa vào DataAdapter

daTheLoai = new SqlDataAdapter(sQueryTheLoai, conn); //Tạo 1 đối tượng SqlDataAdapter tên daNXB

daTheLoai.Fill(ds, "tblTheLoai");

dgTheLoai.DataSource = ds.Tables["tblTheLoai"];

//// … đặt tiêu đề tiếng Việt, định độ rộng cho các trường

dgTheLoai.Columns["MaTheLoai"].HeaderText = "Mã thể loại";

dgTheLoai.Columns["MaTheLoai"].Width = 300;

dgTheLoai.Columns["TenTheLoai"].HeaderText = "Tên thể loại";

dgTheLoai.Columns["TenTheLoai"].Width = 700;

//Command Thêm Thể loại

string sThemTheLoai = @"insert into TheLoai values(@MaTheLoai, @TenTheLoai)";

SqlCommand cmThemTheLoai = new SqlCommand(sThemTheLoai, conn);

cmThemTheLoai.Parameters.Add("@MaTheLoai", SqlDbType.NVarChar, 10, "MaTheLoai");

cmThemTheLoai.Parameters.Add("@TenTheLoai", SqlDbType.NVarChar, 50, "TenTheLoai");

daTheLoai.InsertCommand = cmThemTheLoai;

//Command Sửa Thể loại

string sSuaTheLoai = @"update TheLoai set TenTheLoai=@TenTheLoai where MaTheLoai=@MaTheLoai";

SqlCommand cmSuaTheLoai = new SqlCommand(sSuaTheLoai, conn);

cmSuaTheLoai.Parameters.Add("@MaTheLoai", SqlDbType.NVarChar, 10, "MaTheLoai");

cmSuaTheLoai.Parameters.Add("@TenTheLoai", SqlDbType.NVarChar, 50, "TenTheLoai");

daTheLoai.UpdateCommand = cmSuaTheLoai;

//Command Xóa Thể loại

string sXoaTheLoai = @"delete from TheLoai where MaTheLoai = @MaTheLoai";

SqlCommand cmXoaTheLoai = new SqlCommand(sXoaTheLoai, conn);

cmXoaTheLoai.Parameters.Add("@MaTheLoai", SqlDbType.NVarChar, 5, "MaTheLoai");

daTheLoai.DeleteCommand = cmXoaTheLoai;

}

private void dgTheLoai\_Click(object sender, EventArgs e)

{

DataGridViewRow dr = dgTheLoai.SelectedRows[0];

txtMaTheLoai.Text = dr.Cells["MaTheLoai"].Value.ToString();

txtTenTheLoai.Text = dr.Cells["TenTheLoai"].Value.ToString();

}

private void dgTheLoai\_CellClick(object sender, DataGridViewCellEventArgs e)

{

//Khi chọn 1 dòng trên DataGridView thông tin sẽ hiển thị lên các control tương ứng trên Form

DataGridViewRow dr = dgTheLoai.SelectedRows[0];

txtMaTheLoai.Text = dr.Cells["MaTheLoai"].Value.ToString();

txtTenTheLoai.Text = dr.Cells["TenTheLoai"].Value.ToString();

}

//Kiểm tra Mã NXB trùng

public int kiemtra()

{

for (int i = 0; i < dgTheLoai.RowCount - 1; i++)

{

if (txtMaTheLoai.Text == dgTheLoai.Rows[i].Cells["MaTheLoai"].Value.ToString())

return 0;

}

return 1;

}

private void btnThem\_Click(object sender, EventArgs e)

{

if (txtMaTheLoai.Text == "" || txtTenTheLoai.Text == "")

{

MessageBox.Show("Bạn chưa nhập đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

if (kiemtra() == 0)

MessageBox.Show("Đã tồn tại Mã thể loại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

else

{

//Thêm 1 dòng vào bảng tblSach

DataRow row = ds.Tables["tblTheLoai"].NewRow();

row["MaTheLoai"] = txtMaTheLoai.Text;

row["TenTheLoai"] = txtTenTheLoai.Text;

ds.Tables["tblTheLoai"].Rows.Add(row); //Đưa các dòng này vào table tblTheLoai

}

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

if (txtTenTheLoai.Text == "")

{

MessageBox.Show("Bạn chưa nhập đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

//Thêm 1 dòng vào bảng tblSach

DataGridViewRow dr = dgTheLoai.SelectedRows[0];

dgTheLoai.BeginEdit(true);

dr.Cells["MaTheLoai"].Value = txtMaTheLoai.Text;

dr.Cells["TenTheLoai"].Value = txtTenTheLoai.Text;

dgTheLoai.EndEdit();

MessageBox.Show("Đã cập nhật vào Thông tin Thể loại!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

if (dgTheLoai.SelectedRows.Count > 0) //Lệnh đếm, dòng đã được chọn

{

dgTheLoai.Rows.Remove(dgTheLoai.SelectedRows[0]);

txtMaTheLoai.Clear();

txtTenTheLoai.Clear();

txtMaTheLoai.Focus();

}

}

private void btnLuu\_Click(object sender, EventArgs e)

{

this.daTheLoai.Update(ds, "tblTheLoai");

dgTheLoai.Refresh();

MessageBox.Show("Lưu thành công!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void btnHuy\_Click(object sender, EventArgs e)

{

MessageBox.Show("Bạn có muốn hủy các thao tác vừa rồi không?", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Question);

{

ds.Tables["tbTheLoai"].RejectChanges(); //Hủy các thao tác trước khi lưu

}

}

private void btnThoat\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Bạn có muốn thoát không?", "Thông báo",

MessageBoxButtons.OKCancel, MessageBoxIcon.Question) == DialogResult.OK)

{

Application.Exit();

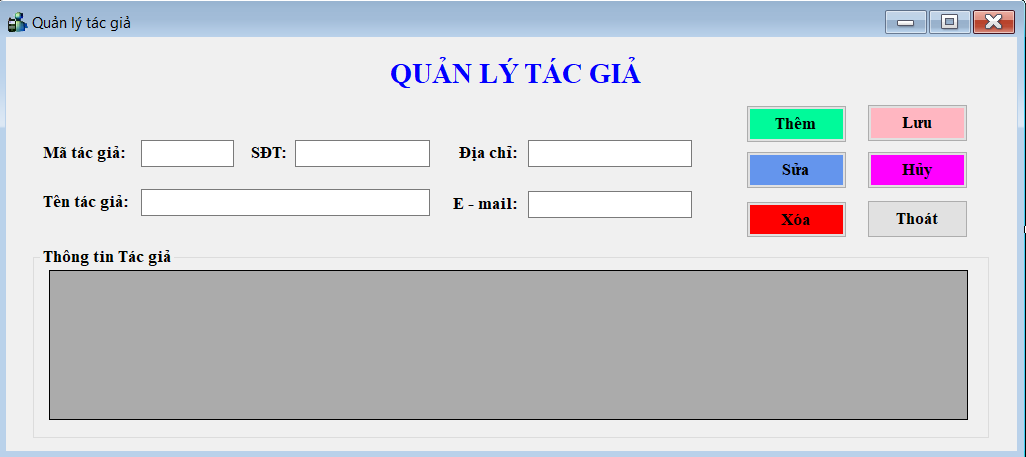
}

}

}

}

**FORM QUẢN LÝ TÁC GIẢ**

****

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace QLSach

{

public partial class frmTacGia : Form

{

public frmTacGia()

{

InitializeComponent();

}

SqlConnection conn = new SqlConnection();

//Khởi tại đối tượng DataSet

DataSet ds = new DataSet();

SqlDataAdapter daTacGia;

private void frmTacGia\_Load(object sender, EventArgs e)

{

conn.ConnectionString = @"Data Source=LAPTOP-3SIUQHOQ\SQL;Initial Catalog=QLSach;Integrated Security=True";

//Dữ liệu Datagridview NXB

string sQueryTacGia = @"select \* from TacGia"; //Câu lệnh truy vấn để đưa vào DataAdapter

daTacGia = new SqlDataAdapter(sQueryTacGia, conn); //Tạo 1 đối tượng SqlDataAdapter tên daTacGia

daTacGia.Fill(ds, "tblTacGia");

dgTacGia.DataSource = ds.Tables["tblTacGia"];

//// … đặt tiêu đề tiếng Việt, định độ rộng cho các trường

dgTacGia.Columns["MaTacGia"].HeaderText = "Mã tác giả";

dgTacGia.Columns["MaTacGia"].Width = 90;

dgTacGia.Columns["TenTacGia"].HeaderText = "Tên tác giả";

dgTacGia.Columns["TenTacGia"].Width = 200;

dgTacGia.Columns["SDT"].HeaderText = "SĐT";

dgTacGia.Columns["SDT"].Width = 150;

dgTacGia.Columns["Email"].HeaderText = "E-mail";

dgTacGia.Columns["Email"].Width = 200;

dgTacGia.Columns["DiaChi"].HeaderText = "Địa chỉ";

dgTacGia.Columns["DiaChi"].Width = 200;

//Command Thêm Tác giả

string sThemTacGia = @"insert into TACGIA values(@MaTacGia, @TenTacGia, @SDT, @Email, @DiaChi)";

SqlCommand cmThemTacGia = new SqlCommand(sThemTacGia, conn);

cmThemTacGia.Parameters.Add("@MaTacGia", SqlDbType.NVarChar, 10, "MaTacGia");

cmThemTacGia.Parameters.Add("@TenTacGia", SqlDbType.NVarChar, 50, "TenTacGia");

cmThemTacGia.Parameters.Add("@SDT", SqlDbType.Int, 50, "SDT");

cmThemTacGia.Parameters.Add("@Email", SqlDbType.NVarChar, 50, "Email");

cmThemTacGia.Parameters.Add("@DiaChi", SqlDbType.NVarChar, 50, "DiaChi");

daTacGia.InsertCommand = cmThemTacGia;

//Command Sửa Tác giả

string sSuaTacGia = @"update TACGIA set TenTacGia=@TenTacGia, SDT=@SDT, Email=@Email, DiaChi=@DiaChi where MaTacGia=@MaTacGia";

SqlCommand cmSuaTacGia = new SqlCommand(sSuaTacGia, conn);

cmSuaTacGia.Parameters.Add("@MaTacGia", SqlDbType.NVarChar, 10, "MaTacGia");

cmSuaTacGia.Parameters.Add("@TenTacGia", SqlDbType.NVarChar, 50, "TenTacGia");

cmSuaTacGia.Parameters.Add("@SDT", SqlDbType.Int, 50, "SDT");

cmSuaTacGia.Parameters.Add("@Email", SqlDbType.NVarChar, 50, "Email");

cmSuaTacGia.Parameters.Add("@DiaChi", SqlDbType.NVarChar, 50, "DiaChi");

daTacGia.UpdateCommand = cmSuaTacGia;

//Command Xóa Tác giả

string sXoaTacGia = @"delete from TACGIA where MaTacGia = @MaTacGia";

SqlCommand cmXoaTacGia = new SqlCommand(sXoaTacGia, conn);

cmXoaTacGia.Parameters.Add("@MaTacGia", SqlDbType.NVarChar, 5, "MaTacGia");

daTacGia.DeleteCommand = cmXoaTacGia;

}

private void dgTacGia\_Click(object sender, EventArgs e)

{

DataGridViewRow dr = dgTacGia.SelectedRows[0];

txtMaTacGia.Text = dr.Cells["MaTacGia"].Value.ToString();

txtTenTacGia.Text = dr.Cells["TenTacGia"].Value.ToString();

txtSDT.Text = dr.Cells["SDT"].Value.ToString();

txtEmail.Text = dr.Cells["Email"].Value.ToString();

txtDiaChi.Text = dr.Cells["DiaChi"].Value.ToString();

}

/\*private void dgTacGia\_CellClick(object sender, DataGridViewCellEventArgs e)

{

//Khi chọn 1 dòng trên DataGridView thông tin sẽ hiển thị lên các control tương ứng trên Form

DataGridViewRow dr = dgTacGia.SelectedRows[0];

txtMaTacGia.Text = dr.Cells["MaTacGia"].Value.ToString();

txtTenTacGia.Text = dr.Cells["TenTacGia"].Value.ToString();

txtSDT.Text = dr.Cells["SDT"].Value.ToString();

txtEmail.Text = dr.Cells["Email"].Value.ToString();

txtDiaChi.Text = dr.Cells["DiaChi"].Value.ToString();

}\*/

public int kiemtra()

{

for (int i = 0; i < dgTacGia.RowCount - 1; i++)

{

if (txtMaTacGia.Text == dgTacGia.Rows[i].Cells["MaTacGia"].Value.ToString())

return 0;

}

return 1;

}

private void btnThem\_Click(object sender, EventArgs e)

{

if (txtMaTacGia.Text == "" || txtTenTacGia.Text == "")

{

MessageBox.Show("Bạn chưa nhập đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

if (kiemtra() == 0)

MessageBox.Show("Đã tồn tại Mã tác giả!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

else

{

//Thêm 1 dòng vào bảng tblSach

DataRow row = ds.Tables["tblTacGia"].NewRow();

row["MaTacGia"] = txtMaTacGia.Text;

row["TenTacGia"] = txtTenTacGia.Text;

row["SDT"] = txtSDT.Text;

row["Email"] = txtEmail.Text;

row["DiaChi"] = txtDiaChi.Text;

ds.Tables["tblTacGia"].Rows.Add(row); //Đưa các dòng này vào table tblTacGia

}

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

if (txtTenTacGia.Text == "")

{

MessageBox.Show("Bạn chưa nhập đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

//Thêm 1 dòng vào bảng tblSach

DataGridViewRow dr = dgTacGia.SelectedRows[0];

dgTacGia.BeginEdit(true);

dr.Cells["MaTacGia"].Value = txtMaTacGia.Text;

dr.Cells["TenTacGia"].Value = txtTenTacGia.Text;

dr.Cells["SDT"].Value = txtSDT.Text;

dr.Cells["Email"].Value = txtEmail.Text;

dr.Cells["DiaChi"].Value = txtDiaChi.Text;

dgTacGia.EndEdit();

MessageBox.Show("Đã cập nhật vào Thông tin Thể loại!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

if (dgTacGia.SelectedRows.Count > 0) //Lệnh đếm, dòng đã được chọn

{

dgTacGia.Rows.Remove(dgTacGia.SelectedRows[0]);

txtMaTacGia.Clear();

txtTenTacGia.Clear();

txtMaTacGia.Focus();

}

}

private void btnLuu\_Click(object sender, EventArgs e)

{

this.daTacGia.Update(ds, "tblTacGia");

dgTacGia.Refresh();

MessageBox.Show("Lưu thành công!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void btnHuy\_Click(object sender, EventArgs e)

{

MessageBox.Show("Bạn có muốn hủy các thao tác vừa rồi không?", "Xác nhận",

MessageBoxButtons.OK, MessageBoxIcon.Question);

{

ds.Tables["tbTacGia"].RejectChanges(); //Hủy các thao tác trước khi lưu

}

}

private void btnThoat\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Bạn có muốn thoát không?", "Xác nhận",

MessageBoxButtons.OKCancel, MessageBoxIcon.Question) == DialogResult.OK)

{

Application.Exit();

}

}

}

}

**FORM QUẢN LÝ NHÀ XUẤT BẢN**

****

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace QLSach

{

public partial class frmNXB : Form

{

public frmNXB()

{

InitializeComponent();

}

SqlConnection conn = new SqlConnection();

//Khởi tại đối tượng DataSet

DataSet ds = new DataSet("dsNXB");

SqlDataAdapter daNXB;

private void frmNXB\_Load(object sender, EventArgs e)

{

conn.ConnectionString = @"Data Source=LAPTOP-3SIUQHOQ\SQL;Initial Catalog=QLSach;Integrated Security=True";

//Dữ liệu Datagridview NXB

string sQueryNXB = @"select n.\* from NXB n"; //Câu lệnh truy vấn để đưa vào DataAdapter

daNXB = new SqlDataAdapter(sQueryNXB, conn); //Tạo 1 đối tượng SqlDataAdapter tên daNXB

daNXB.Fill(ds, "tblNXB");

dgNXB.DataSource = ds.Tables["tblNXB"];

//// … đặt tiêu đề tiếng Việt, định độ rộng cho các trường

dgNXB.Columns["MaNXB"].HeaderText = "Mã NXB";

dgNXB.Columns["MaNXB"].Width = 120;

dgNXB.Columns["TenNXB"].HeaderText = "Tên NXB";

dgNXB.Columns["TenNXB"].Width = 200;

dgNXB.Columns["SDT"].HeaderText = "SĐT";

dgNXB.Columns["SDT"].Width = 150;

dgNXB.Columns["Email"].HeaderText = "E-mail";

dgNXB.Columns["Email"].Width = 280;

dgNXB.Columns["DiaChi"].HeaderText = "Địa chỉ";

dgNXB.Columns["DiaChi"].Width = 250;

//Command Thêm NXB

string sThemNXB = @"insert into NXB values(@MaNXB, @TenNXB, @SDT, @Email, @DiaChi)";

SqlCommand cmThemNXB = new SqlCommand(sThemNXB, conn);

cmThemNXB.Parameters.Add("@MaNXB", SqlDbType.NVarChar, 10, "MaNXB");

cmThemNXB.Parameters.Add("@TenNXB", SqlDbType.NVarChar, 50, "TenNXB");

cmThemNXB.Parameters.Add("@SDT", SqlDbType.Int, 10, "SDT");

cmThemNXB.Parameters.Add("@Email", SqlDbType.NVarChar, 50, "Email");

cmThemNXB.Parameters.Add("@DiaChi", SqlDbType.NVarChar, 50, "DiaChi");

daNXB.InsertCommand = cmThemNXB;

//Command Sửa NXB

string sSuaNXB = @"update NXB set TenNXB=@TenNXB, SDT=@SDT, Email=@Email, DiaChi=@DiaChi where MaNXB=@MaNXB";

SqlCommand cmSuaNXB = new SqlCommand(sSuaNXB, conn);

cmSuaNXB.Parameters.Add("@MaNXB", SqlDbType.NVarChar, 10, "MaNXB");

cmSuaNXB.Parameters.Add("@TenNXB", SqlDbType.NVarChar, 50, "TenNXB");

cmSuaNXB.Parameters.Add("@SDT", SqlDbType.Int, 10, "SDT");

cmSuaNXB.Parameters.Add("@Email", SqlDbType.NVarChar, 50, "Email");

cmSuaNXB.Parameters.Add("@DiaChi", SqlDbType.NVarChar, 50, "DiaChi");

daNXB.UpdateCommand = cmSuaNXB;

//Command Xóa NXB

string sXoaNXB = @"delete from NXB where MaNXB = @MaNXB";

SqlCommand cmXoaNXB = new SqlCommand(sXoaNXB, conn);

cmXoaNXB.Parameters.Add("@MaNXB", SqlDbType.NVarChar, 5, "MaNXB");

daNXB.DeleteCommand = cmXoaNXB;

}

private void dgNXB\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void dgNXB\_Click(object sender, EventArgs e)

{

DataGridViewRow dr = dgNXB.SelectedRows[0];

txtMaNXB.Text = dr.Cells["MaNXB"].Value.ToString();

txtTenNXB.Text = dr.Cells["TenNXB"].Value.ToString();

txtSDT.Text = dr.Cells["SDT"].Value.ToString();

txtEmail.Text = dr.Cells["Email"].Value.ToString();

txtDiaChi.Text = dr.Cells["DiaChi"].Value.ToString();

}

private void dgNXB\_CellClick(object sender, DataGridViewCellEventArgs e)

{

//Khi chọn 1 dòng trên DataGridView thông tin sẽ hiển thị lên các control tương ứng trên Form

DataGridViewRow dr = dgNXB.SelectedRows[0];

txtMaNXB.Text = dr.Cells["MaNXB"].Value.ToString();

txtTenNXB.Text = dr.Cells["TenNXB"].Value.ToString();

txtSDT.Text = dr.Cells["SDT"].Value.ToString();

txtEmail.Text = dr.Cells["Email"].Value.ToString();

txtDiaChi.Text = dr.Cells["DiaChi"].Value.ToString();

}

//Kiểm tra Mã NXB trùng

public int kiemtra()

{

for (int i = 0; i < dgNXB.RowCount - 1; i++)

{

if (txtMaNXB.Text == dgNXB.Rows[i].Cells["MaNXB"].Value.ToString())

return 0;

}

return 1;

}

private void btnThem\_Click(object sender, EventArgs e)

{

if (txtMaNXB.Text == "" || txtTenNXB.Text == "" || txtSDT.Text == "" || txtEmail.Text == "" || txtDiaChi.Text == "")

{

MessageBox.Show("Bạn chưa nhập đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

if (kiemtra() == 0)

MessageBox.Show("Đã tồn tại Mã NXB!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

else

{

//Thêm 1 dòng vào bảng tblSach

DataRow row = ds.Tables["tblNXB"].NewRow();

row["MaNXB"] = txtMaNXB.Text;

row["TenNXB"] = txtTenNXB.Text;

row["SDT"] = txtSDT.Text;

row["Email"] = txtEmail.Text;

row["DiaChi"] = txtDiaChi.Text;

ds.Tables["tblNXB"].Rows.Add(row); //Đưa các dòng này vào table tblNXB

}

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

if (txtTenNXB.Text == "")

{

MessageBox.Show("Bạn chưa nhập đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

//Thêm 1 dòng vào bảng tblNXB

DataGridViewRow dr = dgNXB.SelectedRows[0];

dgNXB.BeginEdit(true);

dr.Cells["MaNXB"].Value = txtMaNXB.Text;

dr.Cells["TenNXB"].Value = txtTenNXB.Text;

dr.Cells["SDT"].Value = txtSDT.Text;

dr.Cells["Email"].Value = txtEmail.Text;

dr.Cells["DiaChi"].Value = txtDiaChi.Text;

dgNXB.EndEdit();

MessageBox.Show("Đã cập nhật vào Thông tin NXB!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

if (dgNXB.SelectedRows.Count > 0) //Lệnh đếm, dòng đã được chọn

{

dgNXB.Rows.Remove(dgNXB.SelectedRows[0]);

txtMaNXB.Clear();

txtTenNXB.Clear();

txtSDT.Clear();

txtEmail.Clear();

txtDiaChi.Clear();

txtMaNXB.Focus();

}

}

private void btnLuu\_Click(object sender, EventArgs e)

{

this.daNXB.Update(ds, "tblNXB");

dgNXB.Refresh();

MessageBox.Show("Lưu thành công!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void btnHuy\_Click(object sender, EventArgs e)

{

MessageBox.Show("Bạn có muốn hủy các thao tác vừa rồi không?", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Question);

{

ds.Tables["tbNXB"].RejectChanges(); //Hủy các thao tác trước khi lưu

}

}

private void btnThoat\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Bạn có muốn thoát không?", "Thông báo",

MessageBoxButtons.OKCancel, MessageBoxIcon.Question) == DialogResult.OK)

{

Application.Exit();

}

}

}

}

**FORM QUẢN LÝ BÁN HÀNG**

****

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace QLSach

{

public partial class frmBanHang : Form

{

public frmBanHang()

{

InitializeComponent();

}

//SqlConnection conn = null;

//Tạo đối tượng conn thuộc lớp SqlConnection

SqlConnection conn = new SqlConnection(@"Data Source=LAPTOP-3SIUQHOQ\SQL;Initial Catalog=QLSach;Integrated Security=True");

//Khởi tại đối tượng DataSet

DataSet ds = new DataSet();

SqlDataAdapter daBanHang;

SqlDataAdapter daSach;

private void frmBanHang\_Load(object sender, EventArgs e)

{

//conn.ConnectionString = @"Data Source=LAPTOP-3SIUQHOQ\SQL;Initial Catalog=QLSach;Integrated Security=True";

//Dữ liệu Datagridview Bán hàng

string sQueryBanHang = @"select b.MaHD, s.MaSach, s.TenSach, b.TGBan, s.GiaBan, b.SLBan, b.ThanhTien, s.SLTon

from BANHANG as b, SACH as s

where b.MaSach = s.MaSach"; //Câu lệnh truy vấn để đưa vào DataAdapter

daBanHang = new SqlDataAdapter(sQueryBanHang, conn); //Tạo 1 đối tượng SqlDataAdapter tên daBanHang

daBanHang.Fill(ds, "tblBanHang");

dgBanHang.DataSource = ds.Tables["tblBanHang"];

//Dữ liệu combobox Sach

string sQuerySach = @"select \* from Sach"; //Câu lệnh truy vấn để đưa vào DataAdapter

daSach = new SqlDataAdapter(sQuerySach, conn); //Tạo 1 đối tượng SqlDataAdapter tên daSach

daSach.Fill(ds, "tblSach");

cmbMaSach.DataSource = ds.Tables["tblSach"]; //Đưa dữ liệu trong DataSet tên ds lên dataGridview

cmbMaSach.DisplayMember = "MaSach"; //Hiện tên trường

cmbMaSach.ValueMember = "MaSach"; //Lấy giá trị của trường

//// … đặt tiêu đề tiếng Việt, định độ rộng cho các trường

dgBanHang.Columns["MaHD"].HeaderText = "Mã HD";

dgBanHang.Columns["MaHD"].Width = 100;

dgBanHang.Columns["MaSach"].HeaderText = "Mã sách";

dgBanHang.Columns["MaSach"].Width = 250;

dgBanHang.Columns["TenSach"].HeaderText = "Tên sách";

dgBanHang.Columns["TenSach"].Width = 300;

dgBanHang.Columns["TGBan"].HeaderText = "Thời gian bán";

dgBanHang.Columns["TGBan"].Width = 150;

dgBanHang.Columns["GiaBan"].HeaderText = "Giá bán";

dgBanHang.Columns["GiaBan"].Width = 120;

dgBanHang.Columns["SLBan"].HeaderText = "SL bán";

dgBanHang.Columns["SLBan"].Width = 100;

dgBanHang.Columns["ThanhTien"].HeaderText = "Thành tiền";

dgBanHang.Columns["ThanhTien"].Width = 120;

dgBanHang.Columns["SLTon"].HeaderText = "SL tồn";

dgBanHang.Columns["SLTon"].Width = 100;

dgBanHang.Columns["MaSach"].Visible = false; //Ẩn cột Mã sách

dgBanHang.Columns["SLTon"].Visible = false;

//Command (hỗ trợ xử lý lệnh) Thêm vào Bán hàng

//Câu lệnh được tham số hóa

string sThemBanHang = @"insert into BANHANG values(@MaHD, @MaSach, @TGBan, @SLBan, @ThanhTien)";

SqlCommand cmThemBanHang = new SqlCommand(sThemBanHang, conn);

cmThemBanHang.Parameters.Add("@MaHD", SqlDbType.NVarChar, 10, "MaHD");

cmThemBanHang.Parameters.Add("@MaSach", SqlDbType.NVarChar, 50, "MaSach");

cmThemBanHang.Parameters.Add("@TGBan", SqlDbType.SmallDateTime, 50, "TGBan");

//cmThemBanHang.Parameters.Add("@GiaBan", SqlDbType.Float, 50, "GiaBan");

cmThemBanHang.Parameters.Add("@SLBan", SqlDbType.Int, 50, "SLBan");

cmThemBanHang.Parameters.Add("@ThanhTien", SqlDbType.Float, 50, "ThanhTien");

daBanHang.InsertCommand = cmThemBanHang;

/\*string sUpdateSLSach = @"update SACH set SLTon = @SLTon where MaSach = @MaSach";

SqlCommand cmUpdateSLSach = new SqlCommand(sUpdateSLSach, conn);

cmUpdateSLSach.Parameters.Add("@SLTon", SqlDbType.Int, 4, "SLTon");

daSach.UpdateCommand = cmUpdateSLSach;\*/

//Command Sửa Bán hàng

string sSuaBanHang = @"update BANHANG set MaSach=@MaSach, TGBan=@TGBan, SLBan=@SLBan, ThanhTien=@ThanhTien

where MaHD=@MaHD";

SqlCommand cmSuaBanHang = new SqlCommand(sSuaBanHang, conn);

cmSuaBanHang.Parameters.Add("@MaHD", SqlDbType.NVarChar, 10, "MaHD");

cmSuaBanHang.Parameters.Add("@MaSach", SqlDbType.NVarChar, 50, "MaSach");

cmSuaBanHang.Parameters.Add("@TGBan", SqlDbType.SmallDateTime, 50, "TGBan");

//cmSuaBanHang.Parameters.Add("@GiaBan", SqlDbType.Float, 50, "GiaBan");

cmSuaBanHang.Parameters.Add("@SLBan", SqlDbType.Int, 50, "SLBan");

cmSuaBanHang.Parameters.Add("@ThanhTien", SqlDbType.Float, 50, "ThanhTien");

daBanHang.UpdateCommand = cmSuaBanHang;

//Command Xóa bán hàng

string sXoaBanHang = @"delete from BANHANG where MaHD = @MaHD";

SqlCommand cmXoaBanHang = new SqlCommand(sXoaBanHang, conn);

cmXoaBanHang.Parameters.Add("@MaHD", SqlDbType.NVarChar, 10, "MaHD");

daBanHang.DeleteCommand = cmXoaBanHang;

}

private void dgBanHang\_CellClick(object sender, DataGridViewCellEventArgs e)

{

//Khi chọn 1 dòng trên DataGridView thông tin sẽ hiển thị lên các control tương ứng trên Form

DataGridViewRow dr = dgBanHang.SelectedRows[0];

txtMaHD.Text = dr.Cells["MaHD"].Value.ToString();

cmbMaSach.Text = dr.Cells["MaSach"].Value.ToString();

txtTenSach.Text = dr.Cells["TenSach"].Value.ToString();

dtTGBan.Text = dr.Cells["TGBan"].Value.ToString();

txtGiaBan.Text = dr.Cells["GiaBan"].Value.ToString();

txtSLBan.Text = dr.Cells["SLBan"].Value.ToString();

txtThanhTien.Text = dr.Cells["ThanhTien"].Value.ToString();

}

//Kiểm tra Mã Hóa đơn trùng

public int kiemtra()

{

for (int i = 0; i < dgBanHang.RowCount - 1; i++)

{

if (txtMaHD.Text == dgBanHang.Rows[i].Cells["MaHD"].Value.ToString())

return 0;

}

return 1;

}

private void btnThem\_Click(object sender, EventArgs e) //Thêm dữ liệu vào dâtgridview

{

if (txtMaHD.Text == "")

{

MessageBox.Show("Bạn chưa nhập Mã Hóa đơn!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

if (kiemtra() == 0)

MessageBox.Show("Đã tồn tại Mã Hóa đơn!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

else

{

//Thêm 1 dòng vào bảng tblBanHang

DataRow row = ds.Tables["tblBanHang"].NewRow();

row["MaHD"] = txtMaHD.Text;

row["MaSach"] = cmbMaSach.SelectedValue;

row["TenSach"] = txtTenSach.Text;

row["TGBan"] = dtTGBan.Text;

row["GiaBan"] = txtGiaBan.Text;

row["SLBan"] = txtSLBan.Text;

row["ThanhTien"] = txtThanhTien.Text;

row["SLTon"] = txtSLTon.Text;

ds.Tables["tblBanHang"].Rows.Add(row); //Đưa các dòng này vào table tblBanHang

}

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

if (txtMaHD.Text == "")

{

MessageBox.Show("Không được chỉnh sửa Mã Hóa đơn!", "Cảnh báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

//Thêm 1 dòng vào bảng tblBanHang

DataGridViewRow dr = dgBanHang.SelectedRows[0];

dgBanHang.BeginEdit(true);

dr.Cells["MaHD"].Value = txtMaHD.Text;

dr.Cells["MaSach"].Value =cmbMaSach.SelectedValue;

dr.Cells["TenSach"].Value = txtTenSach.Text;

dr.Cells["TGBan"].Value = dtTGBan.Text;

dr.Cells["GiaBan"].Value = txtGiaBan.Text;

dr.Cells["SLBan"].Value = txtSLBan.Text;

dr.Cells["ThanhTien"].Value = txtThanhTien.Text;

dgBanHang.EndEdit();

MessageBox.Show("Đã cập nhật vào Thông tin Bán hàng!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

if (dgBanHang.SelectedRows.Count > 0) //Lệnh đếm, dòng đã được chọn

{

MessageBox.Show("Bạn có muốn xóa thông tin này!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

{

dgBanHang.Rows.Remove(dgBanHang.SelectedRows[0]);

txtMaHD.Clear();

cmbMaSach.ResetText();

txtTenSach.ResetText();

dtTGBan.ResetText();

txtGiaBan.Clear();

txtSLBan.Clear();

txtThanhTien.Clear();

txtMaHD.Focus();

}

}

}

private void btnLuu\_Click(object sender, EventArgs e)

{

daBanHang.Update(ds, "tblBanHang");

dgBanHang.Refresh();

MessageBox.Show("Lưu thành công!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void btnHuy\_Click(object sender, EventArgs e)

{

MessageBox.Show("Bạn có muốn hủy các thao tác vừa rồi không?", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Question);

{

ds.Tables["tblBanHang"].RejectChanges(); //Hủy các thao tác trước khi lưu

}

}

private void btnThoat\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Bạn có muốn thoát Quản lý bán hàng không?", "Thông báo",

MessageBoxButtons.OKCancel, MessageBoxIcon.Question) == DialogResult.OK)

{

Application.Exit();

}

}

public string GetFieldValues(string sql) //Truy xuất giá trị hiện tại của một trường

{

string s = "";

SqlCommand cmd = new SqlCommand(sql, conn);

SqlDataReader reader;

reader = cmd.ExecuteReader();

while (reader.Read())

s = reader.GetValue(0).ToString();

reader.Close();

return s;

}

/\*private void cmbTenSach\_TextChanged(object sender, EventArgs e)

{

//if (conn == null)

//conn = new SqlConnection(conn);

if (conn.State == ConnectionState.Closed)

conn.Open();

string str;

if (cmbMaSach.Text == "")

{

txtTenSach.Text = "";

txtGiaBan.Text = "";

txtSLTon.Text = "";

}

//Khi chọn Mã Sách thì các thông tin của Sách sẽ hiện ra

str = "Select TenSach from Sach where MaSach = N'" + cmbMaSach.SelectedValue + "'";

txtTenSach.Text = GetFieldValues(str);

str = "Select GiaBan from Sach where MaSach = N'" + cmbMaSach.SelectedValue + "'";

txtGiaBan.Text = GetFieldValues(str);

str = "Select SLTon from Sach where MaSach = N'" + cmbMaSach.SelectedValue + "'";

txtSLTon.Text = GetFieldValues(str);

conn.Close();

}\*/

private void txtSLBan\_TextChanged(object sender, EventArgs e)

{

double giaban, thanhtien, soluong;

if (txtSLBan.Text == "")

{

soluong = 0;

}

else

{

soluong = Convert.ToDouble(txtSLBan.Text);

giaban = Convert.ToDouble(txtGiaBan.Text);

thanhtien = giaban \* soluong;

txtThanhTien.Text = thanhtien.ToString();

}

double slban, slton;

if (txtSLBan.Text == "")

{

slban = 0;

}

else

{

slban = Convert.ToDouble(txtSLBan.Text);

slton = Convert.ToDouble(txtSLTon.Text);

slton = slton - slban;

txtSLTon.Text = slton.ToString();

}

}

private void cmbMaSach\_TextChanged(object sender, EventArgs e)

{

//if (conn == null)

//conn = new SqlConnection(conn);

if (conn.State == ConnectionState.Closed)

conn.Open();

string str;

if (cmbMaSach.Text == "")

{

txtTenSach.Text = "";

txtGiaBan.Text = "";

txtSLTon.Text = "";

}

//Khi chọn Mã Sách thì các thông tin của Sách sẽ hiện ra

str = "Select TenSach from Sach where MaSach = N'" + cmbMaSach.SelectedValue + "'";

txtTenSach.Text = GetFieldValues(str);

str = "Select GiaBan from Sach where MaSach = N'" + cmbMaSach.SelectedValue + "'";

txtGiaBan.Text = GetFieldValues(str);

str = "Select SLTon from Sach where MaSach = N'" + cmbMaSach.SelectedValue + "'";

txtSLTon.Text = GetFieldValues(str);

conn.Close();

}

}

}

**Phần 5 – CHƯƠNG TRÌNH LÀM ĐƯỢC NHỮNG GÌ?**

* Chương trình thực hiện được các chức năng cơ bản như thêm, xóa, sửa các thông tin của sách, nhà xuất bản, tác giả, thể loại.
* Truy xuất và khai thác được các chức năng chính của phầm mềm như: Quản lý thông tin sách, quản lý bán hàng, và quản lý dữ liệu các form được chặt chẽ cho người dùng.
* Các yêu cầu chưa đạt được:
* Chưa giải quyết được vấn đề khách hàng mua nhiều lúc nhiều quyển sách.
* Số lượng Sách tồn chưa được cập nhật vô cơ sở dữ liệu.

**Phần 6 – TÀI LIỆU THAM KHẢO**

* <https://timoday.edu.vn/phan-4-lap-trinh-co-so-du-lieu-voi-c/>
* <https://timoday.edu.vn/xay-dung-chuong-trinh-quan-ly-ban-hang-bang-c/>
* Giáo trình Lập trình .NET