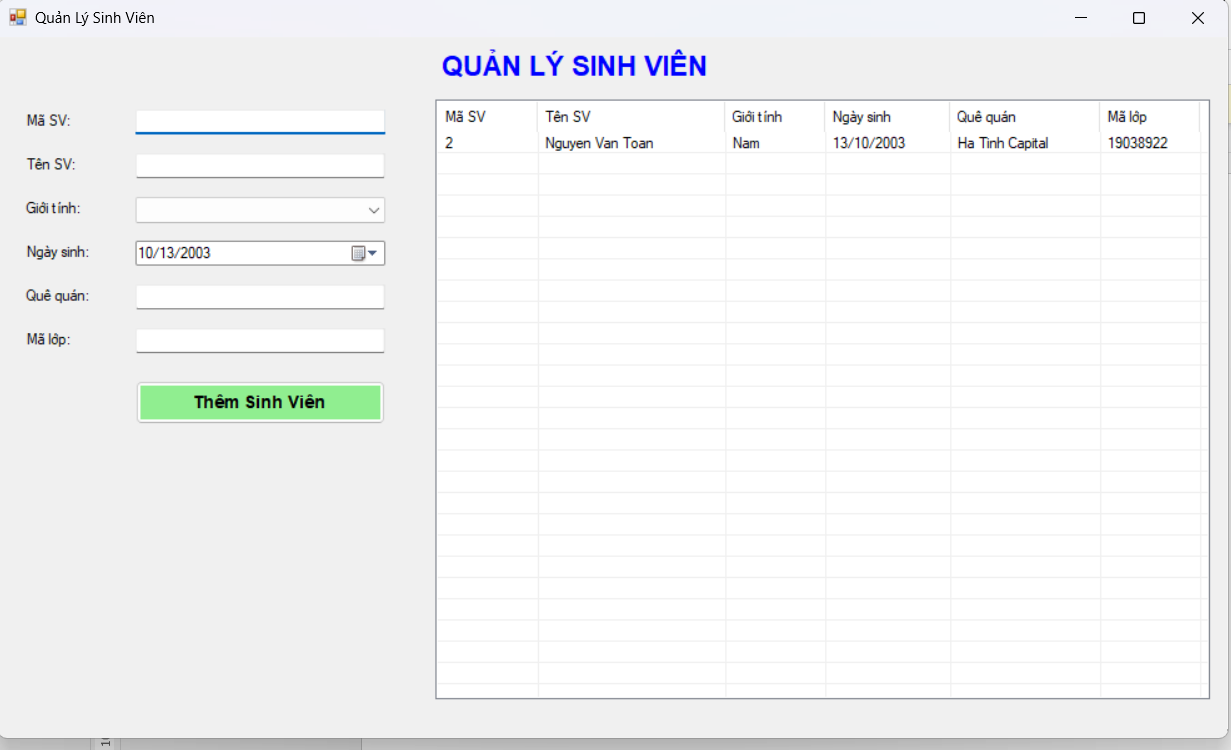
Họ tên: Nguyễn Văn Toàn

MSSV : 1150080040

LAB5

# Thực hành 2

(Thêm dữ liệu chưa có Parameter)



## Code

using System;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Windows.Forms;

namespace QuanLySinhVien

{

public partial class Form1 : Form

{

string strCon = @"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\VanToan123\source\repos\11500800040\_NguyenVanToan\_Lab5\11500800040\_NguyenVanToan\_Lab5\Lab5DB.mdf;Integrated Security=True";

SqlConnection sqlCon = null;

private ListView lsvDanhSachSV;

private TextBox txtMaSV;

private TextBox txtTenSV;

private ComboBox cbGioiTinh;

private DateTimePicker dtpNgaySinh;

private TextBox txtQueQuan;

private TextBox txtMaLop;

private Button btnThemSinhVien;

private Label lblMaSV;

private Label lblTenSV;

private Label lblGioiTinh;

private Label lblNgaySinh;

private Label lblQueQuan;

private Label lblMaLop;

private Label lblTitle;

public Form1()

{

InitializeComponent();

}

private void InitializeComponent()

{

this.Text = "Quản Lý Sinh Viên";

this.Size = new Size(1000, 600);

this.StartPosition = FormStartPosition.CenterScreen;

lblTitle = new Label();

lblTitle.Text = "QUẢN LÝ SINH VIÊN";

lblTitle.Font = new Font("Arial", 16, FontStyle.Bold);

lblTitle.ForeColor = Color.Blue;

lblTitle.Location = new Point(350, 10);

lblTitle.Size = new Size(300, 30);

this.Controls.Add(lblTitle);

lblMaSV = new Label() { Text = "Mã SV:", Location = new Point(20, 60), Size = new Size(80, 20) };

lblTenSV = new Label() { Text = "Tên SV:", Location = new Point(20, 95), Size = new Size(80, 20) };

lblGioiTinh = new Label() { Text = "Giới tính:", Location = new Point(20, 130), Size = new Size(80, 20) };

lblNgaySinh = new Label() { Text = "Ngày sinh:", Location = new Point(20, 165), Size = new Size(80, 20) };

lblQueQuan = new Label() { Text = "Quê quán:", Location = new Point(20, 200), Size = new Size(80, 20) };

lblMaLop = new Label() { Text = "Mã lớp:", Location = new Point(20, 235), Size = new Size(80, 20) };

this.Controls.Add(lblMaSV);

this.Controls.Add(lblTenSV);

this.Controls.Add(lblGioiTinh);

this.Controls.Add(lblNgaySinh);

this.Controls.Add(lblQueQuan);

this.Controls.Add(lblMaLop);

txtMaSV = new TextBox() { Location = new Point(110, 58), Size = new Size(200, 25) };

txtTenSV = new TextBox() { Location = new Point(110, 93), Size = new Size(200, 25) };

txtQueQuan = new TextBox() { Location = new Point(110, 198), Size = new Size(200, 25) };

txtMaLop = new TextBox() { Location = new Point(110, 233), Size = new Size(200, 25) };

this.Controls.Add(txtMaSV);

this.Controls.Add(txtTenSV);

this.Controls.Add(txtQueQuan);

this.Controls.Add(txtMaLop);

cbGioiTinh = new ComboBox();

cbGioiTinh.Location = new Point(110, 128);

cbGioiTinh.Size = new Size(200, 25);

cbGioiTinh.DropDownStyle = ComboBoxStyle.DropDownList;

this.Controls.Add(cbGioiTinh);

dtpNgaySinh = new DateTimePicker();

dtpNgaySinh.Location = new Point(110, 163);

dtpNgaySinh.Size = new Size(200, 25);

dtpNgaySinh.Format = DateTimePickerFormat.Short;

this.Controls.Add(dtpNgaySinh);

btnThemSinhVien = new Button();

btnThemSinhVien.Text = "Thêm Sinh Viên";

btnThemSinhVien.Location = new Point(110, 275);

btnThemSinhVien.Size = new Size(200, 35);

btnThemSinhVien.BackColor = Color.LightGreen;

btnThemSinhVien.Font = new Font("Arial", 10, FontStyle.Bold);

btnThemSinhVien.Click += new EventHandler(btnThemSinhVien\_Click);

this.Controls.Add(btnThemSinhVien);

lsvDanhSachSV = new ListView();

lsvDanhSachSV.Location = new Point(350, 50);

lsvDanhSachSV.Size = new Size(620, 480);

lsvDanhSachSV.View = View.Details;

lsvDanhSachSV.FullRowSelect = true;

lsvDanhSachSV.GridLines = true;

lsvDanhSachSV.Columns.Add("Mã SV", 80);

lsvDanhSachSV.Columns.Add("Tên SV", 150);

lsvDanhSachSV.Columns.Add("Giới tính", 80);

lsvDanhSachSV.Columns.Add("Ngày sinh", 100);

lsvDanhSachSV.Columns.Add("Quê quán", 120);

lsvDanhSachSV.Columns.Add("Mã lớp", 80);

this.Controls.Add(lsvDanhSachSV);

this.Load += new EventHandler(Form1\_Load);

}

private void MoKetNoi()

{

if (sqlCon == null)

{

sqlCon = new SqlConnection(strCon);

}

if (sqlCon.State == ConnectionState.Closed)

{

sqlCon.Open();

}

}

// Hàm đóng kết nối

private void DongKetNoi()

{

if (sqlCon != null && sqlCon.State == ConnectionState.Open)

{

sqlCon.Close();

}

}

// Hàm hiển thị danh sách sinh viên

private void HienThiDanhSach()

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "select \* from SinhVien";

sqlCmd.Connection = sqlCon;

lsvDanhSachSV.Items.Clear();

SqlDataReader reader = sqlCmd.ExecuteReader();

while (reader.Read())

{

string maSv = reader.GetString(0);

string tenSV = reader.GetString(1);

string gioiTinh = reader.GetString(2);

string ngaySinh = reader.GetDateTime(3).ToString("dd/MM/yyyy");

string queQuan = reader.GetString(4);

string maLop = reader.GetString(5);

ListViewItem lvi = new ListViewItem(maSv);

lvi.SubItems.Add(tenSV);

lvi.SubItems.Add(gioiTinh);

lvi.SubItems.Add(ngaySinh);

lvi.SubItems.Add(queQuan);

lvi.SubItems.Add(maLop);

lsvDanhSachSV.Items.Add(lvi);

}

reader.Close();

}

catch (Exception ex)

{

MessageBox.Show("Lỗi hiển thị dữ liệu: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

MoKetNoi();

try

{

string maSV = txtMaSV.Text.Trim();

string tenSV = txtTenSV.Text.Trim();

string gioiTinh = cbGioiTinh.Text;

string ngaySinh = dtpNgaySinh.Value.Month + "/" + dtpNgaySinh.Value.Day + "/" + dtpNgaySinh.Value.Year;

string queQuan = txtQueQuan.Text.Trim();

string maLop = txtMaLop.Text.Trim();

// Kiểm tra dữ liệu đầu vào

if (string.IsNullOrEmpty(maSV) || string.IsNullOrEmpty(tenSV) ||

string.IsNullOrEmpty(gioiTinh) || string.IsNullOrEmpty(queQuan) ||

string.IsNullOrEmpty(maLop))

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "insert into SinhVien values ('" + maSV + "', '" + tenSV + "', '" + gioiTinh + "', '" + ngaySinh + "', '" + queQuan + "', '" + maLop + "')";

sqlCmd.Connection = sqlCon;

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("Thêm sinh viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiDanhSach();

txtMaSV.Clear();

txtTenSV.Clear();

cbGioiTinh.SelectedIndex = -1;

txtQueQuan.Clear();

txtMaLop.Clear();

txtMaSV.Focus();

}

}

catch (Exception ex)

{

MessageBox.Show("Thêm dữ liệu bị lỗi! " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

private void Form1\_Load(object sender, EventArgs e)

{

cbGioiTinh.Items.Add("Nam");

cbGioiTinh.Items.Add("Nữ");

HienThiDanhSach();

}

}

}

Áp dụng 1: Thêm dữ liệu có parameter, thì chỉ cần sửa hàm btnThemSinhVien  
Hàm cũ:

--------------------------------  
private void btnThemSinhVien\_Click(object sender, EventArgs e)

{

MoKetNoi();

try

{

string maSV = txtMaSV.Text.Trim();

string tenSV = txtTenSV.Text.Trim();

string gioiTinh = cbGioiTinh.Text;

string ngaySinh = dtpNgaySinh.Value.Month + "/" + dtpNgaySinh.Value.Day + "/" + dtpNgaySinh.Value.Year;

string queQuan = txtQueQuan.Text.Trim();

string maLop = txtMaLop.Text.Trim();

// Kiểm tra dữ liệu đầu vào

if (string.IsNullOrEmpty(maSV) || string.IsNullOrEmpty(tenSV) ||

string.IsNullOrEmpty(gioiTinh) || string.IsNullOrEmpty(queQuan) ||

string.IsNullOrEmpty(maLop))

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "insert into SinhVien values ('" + maSV + "', '" + tenSV + "', '" + gioiTinh + "', '" + ngaySinh + "', '" + queQuan + "', '" + maLop + "')";

sqlCmd.Connection = sqlCon;

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("Thêm sinh viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiDanhSach();

txtMaSV.Clear();

txtTenSV.Clear();

cbGioiTinh.SelectedIndex = -1;

txtQueQuan.Clear();

txtMaLop.Clear();

txtMaSV.Focus();

}

}

catch (Exception ex)

{

MessageBox.Show("Thêm dữ liệu bị lỗi! " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}  
----------------------------------------

Hàm mới:

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

{

MoKetNoi();

try

{

string maSV = txtMaSV.Text.Trim();

string tenSV = txtTenSV.Text.Trim();

string gioiTinh = cbGioiTinh.Text;

DateTime ngaySinh = dtpNgaySinh.Value;

string queQuan = txtQueQuan.Text.Trim();

string maLop = txtMaLop.Text.Trim();

if (string.IsNullOrEmpty(maSV) || string.IsNullOrEmpty(tenSV) ||

string.IsNullOrEmpty(gioiTinh) || string.IsNullOrEmpty(queQuan) ||

string.IsNullOrEmpty(maLop))

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "INSERT INTO SinhVien (MaSV, TenSV, GioiTinh, NgaySinh, QueQuan, MaLop) " +

"VALUES (@MaSV, @TenSV, @GioiTinh, @NgaySinh, @QueQuan, @MaLop)";

sqlCmd.Connection = sqlCon;

sqlCmd.Parameters.AddWithValue("@MaSV", maSV);

sqlCmd.Parameters.AddWithValue("@TenSV", tenSV);

sqlCmd.Parameters.AddWithValue("@GioiTinh", gioiTinh);

sqlCmd.Parameters.AddWithValue("@NgaySinh", ngaySinh);

sqlCmd.Parameters.AddWithValue("@QueQuan", queQuan);

sqlCmd.Parameters.AddWithValue("@MaLop", maLop);

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("Thêm sinh viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiDanhSach();

txtMaSV.Clear();

txtTenSV.Clear();

cbGioiTinh.SelectedIndex = -1;

txtQueQuan.Clear();

txtMaLop.Clear();

txtMaSV.Focus();

}

}

catch (Exception ex)

{

MessageBox.Show("Thêm dữ liệu bị lỗi! " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

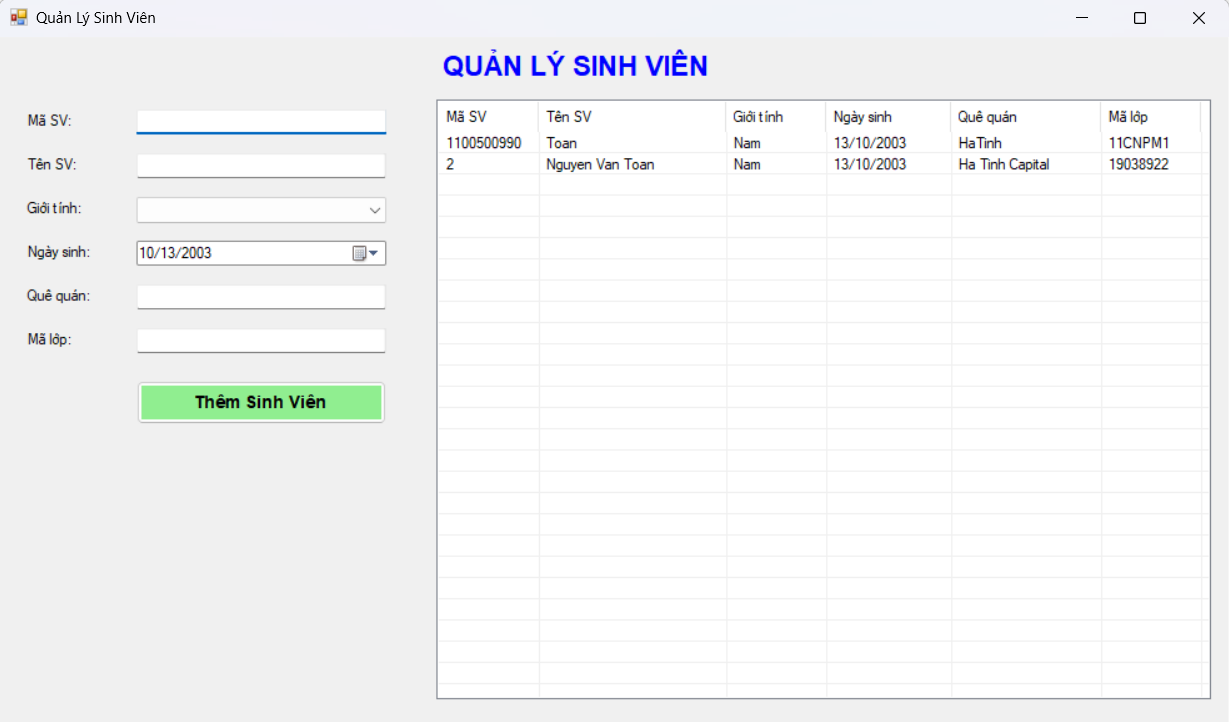
}

finally

{

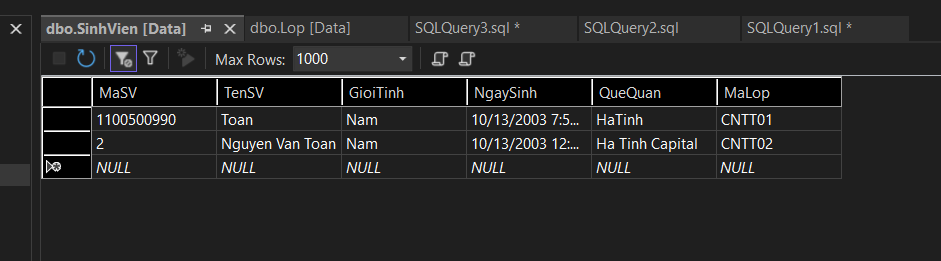
DongKetNoi();

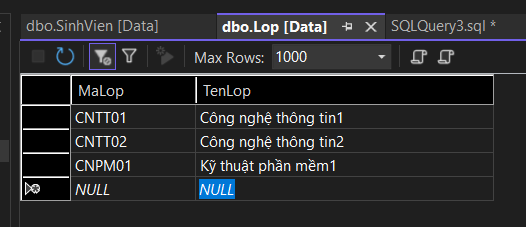
}

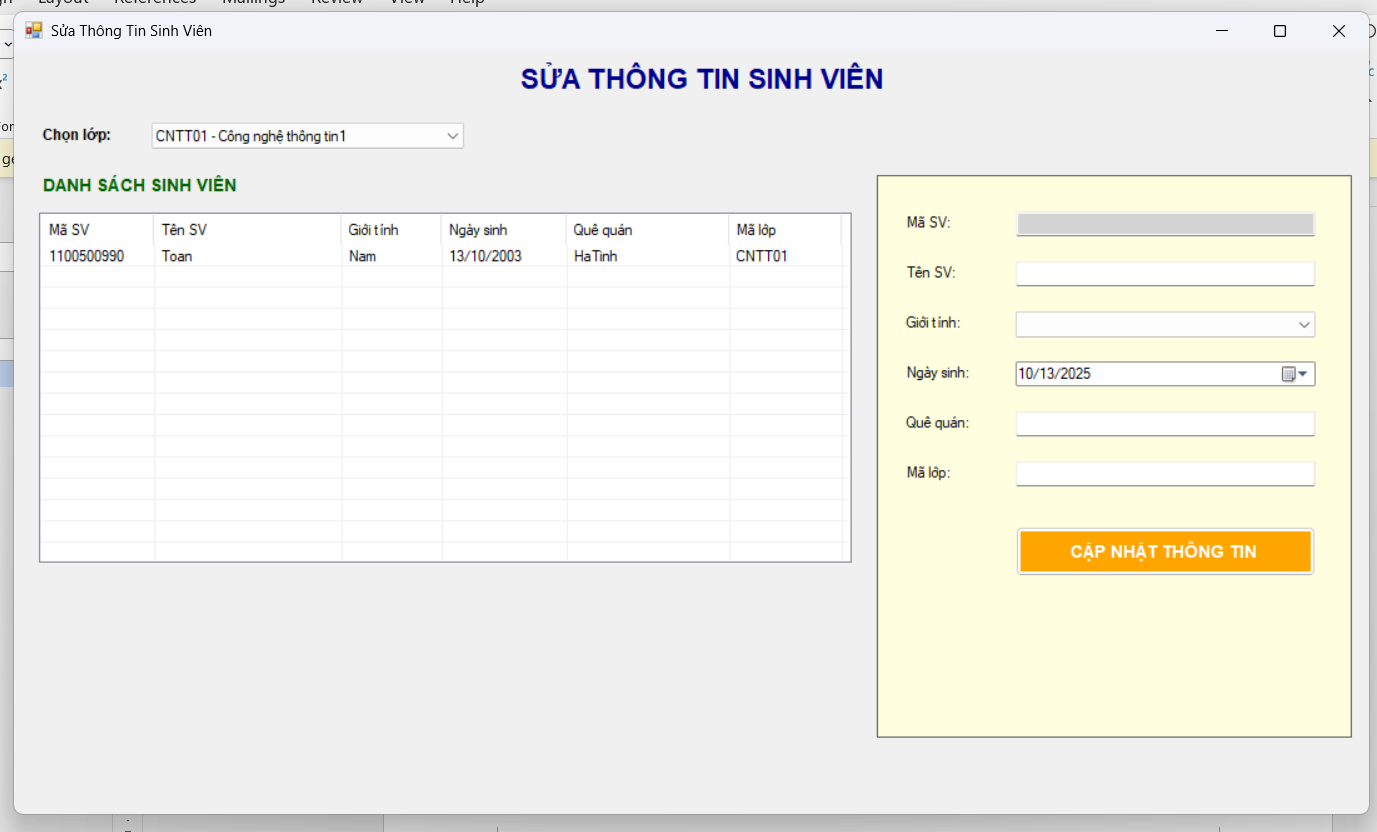
}  


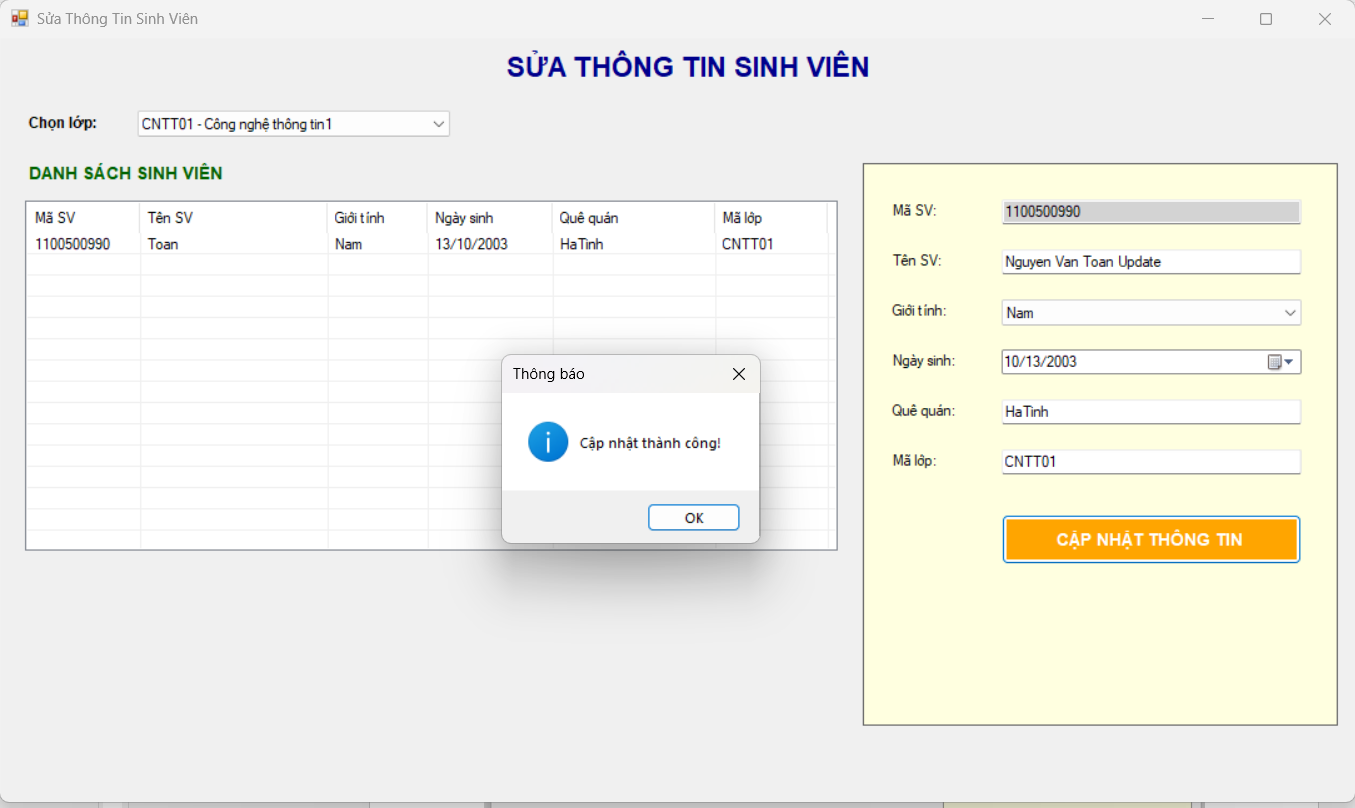
# Thực hành 3

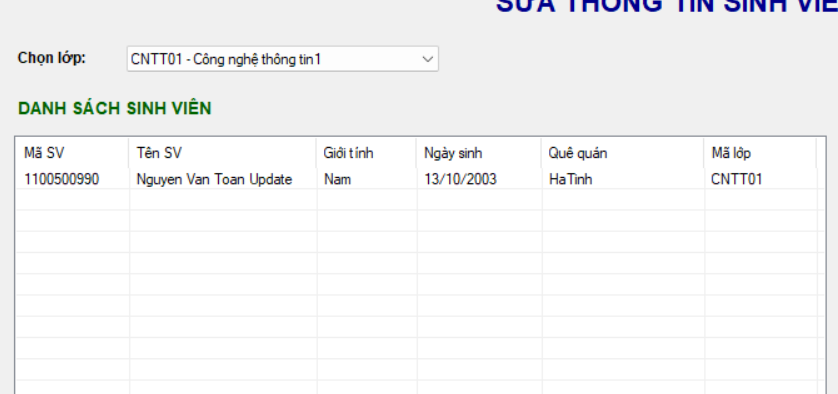
Trước hết, tạo thêm 1 bảng Lop, và phải sửa thông tin MaLop ở bảng SinhVien để nó có dữ liệu tham chiếu











## Code:

using System;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Windows.Forms;

namespace QuanLySinhVien

{

public partial class Form1 : Form

{

string strCon = @"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\VanToan123\source\repos\11500800040\_NguyenVanToan\_Lab5\11500800040\_NguyenVanToan\_Lab5\Lab5DB.mdf;Integrated Security=True";

SqlConnection sqlCon = null;

private ComboBox cbMaLop;

private ListView lsvDanhSach;

private TextBox txtMaSV;

private TextBox txtTenSV;

private ComboBox cbGioiTinh;

private DateTimePicker dtpNgaySinh;

private TextBox txtQueQuan;

private TextBox txtMaLop;

private Button btnSuaThongTin;

private Label lblChonLop;

private Label lblMaSV;

private Label lblTenSV;

private Label lblGioiTinh;

private Label lblNgaySinh;

private Label lblQueQuan;

private Label lblMaLop;

private Label lblTitle;

private Label lblDanhSach;

public Form1()

{

InitializeComponent();

}

private void InitializeComponent()

{

this.Text = "Sửa Thông Tin Sinh Viên";

this.Size = new Size(1100, 650);

this.StartPosition = FormStartPosition.CenterScreen;

// Title

lblTitle = new Label();

lblTitle.Text = "SỬA THÔNG TIN SINH VIÊN";

lblTitle.Font = new Font("Arial", 16, FontStyle.Bold);

lblTitle.ForeColor = Color.DarkBlue;

lblTitle.Location = new Point(400, 10);

lblTitle.Size = new Size(400, 30);

this.Controls.Add(lblTitle);

// Label Chọn lớp

lblChonLop = new Label();

lblChonLop.Text = "Chọn lớp:";

lblChonLop.Location = new Point(20, 60);

lblChonLop.Size = new Size(80, 20);

lblChonLop.Font = new Font("Arial", 9, FontStyle.Bold);

this.Controls.Add(lblChonLop);

// ComboBox Chọn lớp

cbMaLop = new ComboBox();

cbMaLop.Location = new Point(110, 58);

cbMaLop.Size = new Size(250, 25);

cbMaLop.DropDownStyle = ComboBoxStyle.DropDownList;

cbMaLop.SelectedIndexChanged += new EventHandler(cbMaLop\_SelectedIndexChanged);

this.Controls.Add(cbMaLop);

// Label Danh sách

lblDanhSach = new Label();

lblDanhSach.Text = "DANH SÁCH SINH VIÊN";

lblDanhSach.Location = new Point(20, 100);

lblDanhSach.Size = new Size(300, 20);

lblDanhSach.Font = new Font("Arial", 10, FontStyle.Bold);

lblDanhSach.ForeColor = Color.DarkGreen;

this.Controls.Add(lblDanhSach);

// ListView Danh sách sinh viên

lsvDanhSach = new ListView();

lsvDanhSach.Location = new Point(20, 130);

lsvDanhSach.Size = new Size(650, 280);

lsvDanhSach.View = View.Details;

lsvDanhSach.FullRowSelect = true;

lsvDanhSach.GridLines = true;

lsvDanhSach.Columns.Add("Mã SV", 90);

lsvDanhSach.Columns.Add("Tên SV", 150);

lsvDanhSach.Columns.Add("Giới tính", 80);

lsvDanhSach.Columns.Add("Ngày sinh", 100);

lsvDanhSach.Columns.Add("Quê quán", 130);

lsvDanhSach.Columns.Add("Mã lớp", 90);

lsvDanhSach.SelectedIndexChanged += new EventHandler(lsvDanhSach\_SelectedIndexChanged);

this.Controls.Add(lsvDanhSach);

// Panel chứa form sửa thông tin

Panel panelSua = new Panel();

panelSua.Location = new Point(690, 100);

panelSua.Size = new Size(380, 450);

panelSua.BorderStyle = BorderStyle.FixedSingle;

panelSua.BackColor = Color.LightYellow;

// Labels trong panel

lblMaSV = new Label() { Text = "Mã SV:", Location = new Point(20, 30), Size = new Size(80, 20) };

lblTenSV = new Label() { Text = "Tên SV:", Location = new Point(20, 70), Size = new Size(80, 20) };

lblGioiTinh = new Label() { Text = "Giới tính:", Location = new Point(20, 110), Size = new Size(80, 20) };

lblNgaySinh = new Label() { Text = "Ngày sinh:", Location = new Point(20, 150), Size = new Size(80, 20) };

lblQueQuan = new Label() { Text = "Quê quán:", Location = new Point(20, 190), Size = new Size(80, 20) };

lblMaLop = new Label() { Text = "Mã lớp:", Location = new Point(20, 230), Size = new Size(80, 20) };

panelSua.Controls.Add(lblMaSV);

panelSua.Controls.Add(lblTenSV);

panelSua.Controls.Add(lblGioiTinh);

panelSua.Controls.Add(lblNgaySinh);

panelSua.Controls.Add(lblQueQuan);

panelSua.Controls.Add(lblMaLop);

// TextBoxes trong panel

txtMaSV = new TextBox() { Location = new Point(110, 28), Size = new Size(240, 25), ReadOnly = true, BackColor = Color.LightGray };

txtTenSV = new TextBox() { Location = new Point(110, 68), Size = new Size(240, 25) };

txtQueQuan = new TextBox() { Location = new Point(110, 188), Size = new Size(240, 25) };

txtMaLop = new TextBox() { Location = new Point(110, 228), Size = new Size(240, 25) };

panelSua.Controls.Add(txtMaSV);

panelSua.Controls.Add(txtTenSV);

panelSua.Controls.Add(txtQueQuan);

panelSua.Controls.Add(txtMaLop);

// ComboBox Giới tính trong panel

cbGioiTinh = new ComboBox();

cbGioiTinh.Location = new Point(110, 108);

cbGioiTinh.Size = new Size(240, 25);

cbGioiTinh.DropDownStyle = ComboBoxStyle.DropDownList;

cbGioiTinh.Items.Add("Nam");

cbGioiTinh.Items.Add("Nữ");

panelSua.Controls.Add(cbGioiTinh);

// DateTimePicker trong panel

dtpNgaySinh = new DateTimePicker();

dtpNgaySinh.Location = new Point(110, 148);

dtpNgaySinh.Size = new Size(240, 25);

dtpNgaySinh.Format = DateTimePickerFormat.Short;

panelSua.Controls.Add(dtpNgaySinh);

// Button Sửa trong panel

btnSuaThongTin = new Button();

btnSuaThongTin.Text = "CẬP NHẬT THÔNG TIN";

btnSuaThongTin.Location = new Point(110, 280);

btnSuaThongTin.Size = new Size(240, 40);

btnSuaThongTin.BackColor = Color.Orange;

btnSuaThongTin.Font = new Font("Arial", 10, FontStyle.Bold);

btnSuaThongTin.ForeColor = Color.White;

btnSuaThongTin.Click += new EventHandler(btnSuaThongTin\_Click);

panelSua.Controls.Add(btnSuaThongTin);

this.Controls.Add(panelSua);

// Load event

this.Load += new EventHandler(Form1\_Load);

}

// Hàm mở kết nối

private void MoKetNoi()

{

if (sqlCon == null)

{

sqlCon = new SqlConnection(strCon);

}

if (sqlCon.State == ConnectionState.Closed)

{

sqlCon.Open();

}

}

// Hàm đóng kết nối

private void DongKetNoi()

{

if (sqlCon != null && sqlCon.State == ConnectionState.Open)

{

sqlCon.Close();

}

}

// Hàm hiển thị danh sách mã lớp

private void HienThiDSMaLop()

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "select \* from Lop";

sqlCmd.Connection = sqlCon;

SqlDataReader reader = sqlCmd.ExecuteReader();

cbMaLop.Items.Clear();

while (reader.Read())

{

string maLop = reader.GetString(0);

string tenLop = reader.GetString(1);

cbMaLop.Items.Add(maLop + " - " + tenLop);

}

reader.Close();

}

catch (Exception ex)

{

MessageBox.Show("Lỗi hiển thị danh sách lớp: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

// Hàm hiển thị danh sách sinh viên theo lớp đã chọn

private void HienThiDSinhVienTheoLop(string maLop)

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "select \* from SinhVien where MaLop='" + maLop + "'";

sqlCmd.Connection = sqlCon;

lsvDanhSach.Items.Clear();

SqlDataReader reader = sqlCmd.ExecuteReader();

while (reader.Read())

{

string maSV = reader.GetString(0);

string tenSV = reader.GetString(1);

string gioiTinh = reader.GetString(2);

string ngaySinh = reader.GetDateTime(3).ToString("dd/MM/yyyy");

string queQuan = reader.GetString(4);

string maLopSV = reader.GetString(5);

ListViewItem lvi = new ListViewItem(maSV);

lvi.SubItems.Add(tenSV);

lvi.SubItems.Add(gioiTinh);

lvi.SubItems.Add(ngaySinh);

lvi.SubItems.Add(queQuan);

lvi.SubItems.Add(maLopSV);

lsvDanhSach.Items.Add(lvi);

}

reader.Close();

}

catch (Exception ex)

{

MessageBox.Show("Lỗi hiển thị danh sách sinh viên: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

// Sự kiện Load form

private void Form1\_Load(object sender, EventArgs e)

{

HienThiDSMaLop();

}

// Sự kiện khi chọn lớp

private void cbMaLop\_SelectedIndexChanged(object sender, EventArgs e)

{

// Nếu chưa chọn lớp nào

if (cbMaLop.SelectedIndex == -1) return;

// Nếu đã chọn 1 lớp

string[] line = cbMaLop.SelectedItem.ToString().Split('-');

string maLop = line[0].Trim();

// Hiển thị thông tin sinh viên theo mã lớp đã chọn

HienThiDSinhVienTheoLop(maLop);

}

// Sự kiện khi chọn sinh viên trong ListView

private void lsvDanhSach\_SelectedIndexChanged(object sender, EventArgs e)

{

// Nếu chưa chọn sinh viên nào

if (lsvDanhSach.SelectedItems.Count == 0) return;

// Nếu đã chọn 1 sinh viên

ListViewItem lvi = lsvDanhSach.SelectedItems[0];

// Hiển thị thông tin lên các control

txtMaSV.Text = lvi.SubItems[0].Text;

txtTenSV.Text = lvi.SubItems[1].Text;

string gioiTinh = lvi.SubItems[2].Text;

cbGioiTinh.SelectedIndex = (gioiTinh == "Nam") ? 0 : 1;

string[] dt = lvi.SubItems[3].Text.Split('/');

dtpNgaySinh.Value = new DateTime(

int.Parse(dt[2].Trim()),

int.Parse(dt[1].Trim()),

int.Parse(dt[0].Trim())

);

txtQueQuan.Text = lvi.SubItems[4].Text;

txtMaLop.Text = lvi.SubItems[5].Text;

}

// Sự kiện sửa thông tin sinh viên

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

{

try

{

// Kiểm tra xem đã chọn sinh viên chưa

if (string.IsNullOrEmpty(txtMaSV.Text))

{

MessageBox.Show("Vui lòng chọn sinh viên cần sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

MoKetNoi();

// Lấy dữ liệu từ control

string maSV = txtMaSV.Text;

string tenSV = txtTenSV.Text;

string gioiTinh = cbGioiTinh.Text;

string ngaySinh = dtpNgaySinh.Value.ToString("yyyy/MM/dd");

string queQuan = txtQueQuan.Text;

string maLop = txtMaLop.Text;

// Kiểm tra dữ liệu

if (string.IsNullOrEmpty(tenSV) || string.IsNullOrEmpty(gioiTinh) ||

string.IsNullOrEmpty(queQuan) || string.IsNullOrEmpty(maLop))

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "update SinhVien set MaSV='" + maSV + "',TenSV='" + tenSV +

"',GioiTinh='" + gioiTinh + "',NgaySinh=Cast('" + ngaySinh +

"' as Datetime),QueQuan='" + queQuan + "',MaLop='" + maLop +

"' where MaSV='" + maSV + "'";

sqlCmd.Connection = sqlCon;

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("Cập nhật thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiDSinhVienTheoLop(maLop);

}

else

{

MessageBox.Show("Cập nhật không thành công!", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

// Xóa dữ liệu trên form

txtMaSV.Text = txtTenSV.Text = txtQueQuan.Text = txtMaLop.Text = "";

cbGioiTinh.SelectedIndex = -1;

}

catch (Exception ex)

{

MessageBox.Show("Lỗi cập nhật: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

}

}

## Áp dụng 2:

Hàm sửa thông tin cũ:

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

{

try

{

if (string.IsNullOrEmpty(txtMaSV.Text))

{

MessageBox.Show("Vui lòng chọn sinh viên cần sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

MoKetNoi();

string maSV = txtMaSV.Text;

string tenSV = txtTenSV.Text;

string gioiTinh = cbGioiTinh.Text;

string ngaySinh = dtpNgaySinh.Value.ToString("yyyy/MM/dd");

string queQuan = txtQueQuan.Text;

string maLop = txtMaLop.Text;

if (string.IsNullOrEmpty(tenSV) || string.IsNullOrEmpty(gioiTinh) ||

string.IsNullOrEmpty(queQuan) || string.IsNullOrEmpty(maLop))

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "update SinhVien set MaSV='" + maSV + "',TenSV='" + tenSV +

"',GioiTinh='" + gioiTinh + "',NgaySinh=Cast('" + ngaySinh +

"' as Datetime),QueQuan='" + queQuan + "',MaLop='" + maLop +

"' where MaSV='" + maSV + "'";

sqlCmd.Connection = sqlCon;

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("Cập nhật thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiDSinhVienTheoLop(maLop);

}

else

{

MessageBox.Show("Cập nhật không thành công!", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

txtMaSV.Text = txtTenSV.Text = txtQueQuan.Text = txtMaLop.Text = "";

cbGioiTinh.SelectedIndex = -1;

}

catch (Exception ex)

{

MessageBox.Show("Lỗi cập nhật: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}  
  
----------------------------------------

Hàm sửa thông tin mới:

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

{

try

{

// Kiểm tra xem đã chọn sinh viên chưa

if (string.IsNullOrEmpty(txtMaSV.Text))

{

MessageBox.Show("Vui lòng chọn sinh viên cần sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

MoKetNoi();

// Lấy dữ liệu từ control

string maSV = txtMaSV.Text.Trim();

string tenSV = txtTenSV.Text.Trim();

string gioiTinh = cbGioiTinh.Text.Trim();

DateTime ngaySinh = dtpNgaySinh.Value;

string queQuan = txtQueQuan.Text.Trim();

string maLop = txtMaLop.Text.Trim();

// Kiểm tra dữ liệu

if (string.IsNullOrEmpty(tenSV) || string.IsNullOrEmpty(gioiTinh) ||

string.IsNullOrEmpty(queQuan) || string.IsNullOrEmpty(maLop))

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

// Sử dụng Parameter để tránh SQL Injection

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = @"

UPDATE SinhVien

SET TenSV = @TenSV,

GioiTinh = @GioiTinh,

NgaySinh = @NgaySinh,

QueQuan = @QueQuan,

MaLop = @MaLop

WHERE MaSV = @MaSV";

sqlCmd.Connection = sqlCon;

// Thêm các parameter vào lệnh SQL

sqlCmd.Parameters.AddWithValue("@MaSV", maSV);

sqlCmd.Parameters.AddWithValue("@TenSV", tenSV);

sqlCmd.Parameters.AddWithValue("@GioiTinh", gioiTinh);

sqlCmd.Parameters.AddWithValue("@NgaySinh", ngaySinh);

sqlCmd.Parameters.AddWithValue("@QueQuan", queQuan);

sqlCmd.Parameters.AddWithValue("@MaLop", maLop);

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("Cập nhật thông tin sinh viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiDSinhVienTheoLop(maLop);

}

else

{

MessageBox.Show("Không có bản ghi nào được cập nhật!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

// Xóa dữ liệu trên form

txtMaSV.Clear();

txtTenSV.Clear();

txtQueQuan.Clear();

txtMaLop.Clear();

cbGioiTinh.SelectedIndex = -1;

}

catch (Exception ex)

{

MessageBox.Show("Lỗi cập nhật: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

finally

{

DongKetNoi();

}

}

Full code của Thực hành 3 + áp dụng 2

using System;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Windows.Forms;

namespace QuanLySinhVien

{

public partial class Form1 : Form

{

string strCon = @"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\VanToan123\source\repos\11500800040\_NguyenVanToan\_Lab5\11500800040\_NguyenVanToan\_Lab5\Lab5DB.mdf;Integrated Security=True";

SqlConnection sqlCon = null;

private ComboBox cbMaLop;

private ListView lsvDanhSach;

private TextBox txtMaSV;

private TextBox txtTenSV;

private ComboBox cbGioiTinh;

private DateTimePicker dtpNgaySinh;

private TextBox txtQueQuan;

private TextBox txtMaLop;

private Button btnSuaThongTin;

private Label lblChonLop;

private Label lblMaSV;

private Label lblTenSV;

private Label lblGioiTinh;

private Label lblNgaySinh;

private Label lblQueQuan;

private Label lblMaLop;

private Label lblTitle;

private Label lblDanhSach;

public Form1()

{

InitializeComponent();

}

private void InitializeComponent()

{

this.Text = "Sửa Thông Tin Sinh Viên";

this.Size = new Size(1100, 650);

this.StartPosition = FormStartPosition.CenterScreen;

// Title

lblTitle = new Label();

lblTitle.Text = "SỬA THÔNG TIN SINH VIÊN";

lblTitle.Font = new Font("Arial", 16, FontStyle.Bold);

lblTitle.ForeColor = Color.DarkBlue;

lblTitle.Location = new Point(400, 10);

lblTitle.Size = new Size(400, 30);

this.Controls.Add(lblTitle);

// Label Chọn lớp

lblChonLop = new Label();

lblChonLop.Text = "Chọn lớp:";

lblChonLop.Location = new Point(20, 60);

lblChonLop.Size = new Size(80, 20);

lblChonLop.Font = new Font("Arial", 9, FontStyle.Bold);

this.Controls.Add(lblChonLop);

// ComboBox Chọn lớp

cbMaLop = new ComboBox();

cbMaLop.Location = new Point(110, 58);

cbMaLop.Size = new Size(250, 25);

cbMaLop.DropDownStyle = ComboBoxStyle.DropDownList;

cbMaLop.SelectedIndexChanged += new EventHandler(cbMaLop\_SelectedIndexChanged);

this.Controls.Add(cbMaLop);

// Label Danh sách

lblDanhSach = new Label();

lblDanhSach.Text = "DANH SÁCH SINH VIÊN";

lblDanhSach.Location = new Point(20, 100);

lblDanhSach.Size = new Size(300, 20);

lblDanhSach.Font = new Font("Arial", 10, FontStyle.Bold);

lblDanhSach.ForeColor = Color.DarkGreen;

this.Controls.Add(lblDanhSach);

// ListView Danh sách sinh viên

lsvDanhSach = new ListView();

lsvDanhSach.Location = new Point(20, 130);

lsvDanhSach.Size = new Size(650, 280);

lsvDanhSach.View = View.Details;

lsvDanhSach.FullRowSelect = true;

lsvDanhSach.GridLines = true;

lsvDanhSach.Columns.Add("Mã SV", 90);

lsvDanhSach.Columns.Add("Tên SV", 150);

lsvDanhSach.Columns.Add("Giới tính", 80);

lsvDanhSach.Columns.Add("Ngày sinh", 100);

lsvDanhSach.Columns.Add("Quê quán", 130);

lsvDanhSach.Columns.Add("Mã lớp", 90);

lsvDanhSach.SelectedIndexChanged += new EventHandler(lsvDanhSach\_SelectedIndexChanged);

this.Controls.Add(lsvDanhSach);

// Panel chứa form sửa thông tin

Panel panelSua = new Panel();

panelSua.Location = new Point(690, 100);

panelSua.Size = new Size(380, 450);

panelSua.BorderStyle = BorderStyle.FixedSingle;

panelSua.BackColor = Color.LightYellow;

// Labels trong panel

lblMaSV = new Label() { Text = "Mã SV:", Location = new Point(20, 30), Size = new Size(80, 20) };

lblTenSV = new Label() { Text = "Tên SV:", Location = new Point(20, 70), Size = new Size(80, 20) };

lblGioiTinh = new Label() { Text = "Giới tính:", Location = new Point(20, 110), Size = new Size(80, 20) };

lblNgaySinh = new Label() { Text = "Ngày sinh:", Location = new Point(20, 150), Size = new Size(80, 20) };

lblQueQuan = new Label() { Text = "Quê quán:", Location = new Point(20, 190), Size = new Size(80, 20) };

lblMaLop = new Label() { Text = "Mã lớp:", Location = new Point(20, 230), Size = new Size(80, 20) };

panelSua.Controls.Add(lblMaSV);

panelSua.Controls.Add(lblTenSV);

panelSua.Controls.Add(lblGioiTinh);

panelSua.Controls.Add(lblNgaySinh);

panelSua.Controls.Add(lblQueQuan);

panelSua.Controls.Add(lblMaLop);

// TextBoxes trong panel

txtMaSV = new TextBox() { Location = new Point(110, 28), Size = new Size(240, 25), ReadOnly = true, BackColor = Color.LightGray };

txtTenSV = new TextBox() { Location = new Point(110, 68), Size = new Size(240, 25) };

txtQueQuan = new TextBox() { Location = new Point(110, 188), Size = new Size(240, 25) };

txtMaLop = new TextBox() { Location = new Point(110, 228), Size = new Size(240, 25) };

panelSua.Controls.Add(txtMaSV);

panelSua.Controls.Add(txtTenSV);

panelSua.Controls.Add(txtQueQuan);

panelSua.Controls.Add(txtMaLop);

// ComboBox Giới tính trong panel

cbGioiTinh = new ComboBox();

cbGioiTinh.Location = new Point(110, 108);

cbGioiTinh.Size = new Size(240, 25);

cbGioiTinh.DropDownStyle = ComboBoxStyle.DropDownList;

cbGioiTinh.Items.Add("Nam");

cbGioiTinh.Items.Add("Nữ");

panelSua.Controls.Add(cbGioiTinh);

// DateTimePicker trong panel

dtpNgaySinh = new DateTimePicker();

dtpNgaySinh.Location = new Point(110, 148);

dtpNgaySinh.Size = new Size(240, 25);

dtpNgaySinh.Format = DateTimePickerFormat.Short;

panelSua.Controls.Add(dtpNgaySinh);

// Button Sửa trong panel

btnSuaThongTin = new Button();

btnSuaThongTin.Text = "CẬP NHẬT THÔNG TIN";

btnSuaThongTin.Location = new Point(110, 280);

btnSuaThongTin.Size = new Size(240, 40);

btnSuaThongTin.BackColor = Color.Orange;

btnSuaThongTin.Font = new Font("Arial", 10, FontStyle.Bold);

btnSuaThongTin.ForeColor = Color.White;

btnSuaThongTin.Click += new EventHandler(btnSuaThongTin\_Click);

panelSua.Controls.Add(btnSuaThongTin);

this.Controls.Add(panelSua);

// Load event

this.Load += new EventHandler(Form1\_Load);

}

// Hàm mở kết nối

private void MoKetNoi()

{

if (sqlCon == null)

{

sqlCon = new SqlConnection(strCon);

}

if (sqlCon.State == ConnectionState.Closed)

{

sqlCon.Open();

}

}

// Hàm đóng kết nối

private void DongKetNoi()

{

if (sqlCon != null && sqlCon.State == ConnectionState.Open)

{

sqlCon.Close();

}

}

// Hàm hiển thị danh sách mã lớp

private void HienThiDSMaLop()

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "select \* from Lop";

sqlCmd.Connection = sqlCon;

SqlDataReader reader = sqlCmd.ExecuteReader();

cbMaLop.Items.Clear();

while (reader.Read())

{

string maLop = reader.GetString(0);

string tenLop = reader.GetString(1);

cbMaLop.Items.Add(maLop + " - " + tenLop);

}

reader.Close();

}

catch (Exception ex)

{

MessageBox.Show("Lỗi hiển thị danh sách lớp: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

// Hàm hiển thị danh sách sinh viên theo lớp đã chọn

private void HienThiDSinhVienTheoLop(string maLop)

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "select \* from SinhVien where MaLop='" + maLop + "'";

sqlCmd.Connection = sqlCon;

lsvDanhSach.Items.Clear();

SqlDataReader reader = sqlCmd.ExecuteReader();

while (reader.Read())

{

string maSV = reader.GetString(0);

string tenSV = reader.GetString(1);

string gioiTinh = reader.GetString(2);

string ngaySinh = reader.GetDateTime(3).ToString("dd/MM/yyyy");

string queQuan = reader.GetString(4);

string maLopSV = reader.GetString(5);

ListViewItem lvi = new ListViewItem(maSV);

lvi.SubItems.Add(tenSV);

lvi.SubItems.Add(gioiTinh);

lvi.SubItems.Add(ngaySinh);

lvi.SubItems.Add(queQuan);

lvi.SubItems.Add(maLopSV);

lsvDanhSach.Items.Add(lvi);

}

reader.Close();

}

catch (Exception ex)

{

MessageBox.Show("Lỗi hiển thị danh sách sinh viên: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

// Sự kiện Load form

private void Form1\_Load(object sender, EventArgs e)

{

HienThiDSMaLop();

}

// Sự kiện khi chọn lớp

private void cbMaLop\_SelectedIndexChanged(object sender, EventArgs e)

{

// Nếu chưa chọn lớp nào

if (cbMaLop.SelectedIndex == -1) return;

// Nếu đã chọn 1 lớp

string[] line = cbMaLop.SelectedItem.ToString().Split('-');

string maLop = line[0].Trim();

// Hiển thị thông tin sinh viên theo mã lớp đã chọn

HienThiDSinhVienTheoLop(maLop);

}

// Sự kiện khi chọn sinh viên trong ListView

private void lsvDanhSach\_SelectedIndexChanged(object sender, EventArgs e)

{

// Nếu chưa chọn sinh viên nào

if (lsvDanhSach.SelectedItems.Count == 0) return;

// Nếu đã chọn 1 sinh viên

ListViewItem lvi = lsvDanhSach.SelectedItems[0];

// Hiển thị thông tin lên các control

txtMaSV.Text = lvi.SubItems[0].Text;

txtTenSV.Text = lvi.SubItems[1].Text;

string gioiTinh = lvi.SubItems[2].Text;

cbGioiTinh.SelectedIndex = (gioiTinh == "Nam") ? 0 : 1;

string[] dt = lvi.SubItems[3].Text.Split('/');

dtpNgaySinh.Value = new DateTime(

int.Parse(dt[2].Trim()),

int.Parse(dt[1].Trim()),

int.Parse(dt[0].Trim())

);

txtQueQuan.Text = lvi.SubItems[4].Text;

txtMaLop.Text = lvi.SubItems[5].Text;

}

// Sự kiện sửa thông tin sinh viên

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

{

try

{

// Kiểm tra xem đã chọn sinh viên chưa

if (string.IsNullOrEmpty(txtMaSV.Text))

{

MessageBox.Show("Vui lòng chọn sinh viên cần sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

MoKetNoi();

// Lấy dữ liệu từ control

string maSV = txtMaSV.Text.Trim();

string tenSV = txtTenSV.Text.Trim();

string gioiTinh = cbGioiTinh.Text.Trim();

DateTime ngaySinh = dtpNgaySinh.Value;

string queQuan = txtQueQuan.Text.Trim();

string maLop = txtMaLop.Text.Trim();

// Kiểm tra dữ liệu

if (string.IsNullOrEmpty(tenSV) || string.IsNullOrEmpty(gioiTinh) ||

string.IsNullOrEmpty(queQuan) || string.IsNullOrEmpty(maLop))

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

return;

}

// Sử dụng Parameter để tránh SQL Injection

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = @"

UPDATE SinhVien

SET TenSV = @TenSV,

GioiTinh = @GioiTinh,

NgaySinh = @NgaySinh,

QueQuan = @QueQuan,

MaLop = @MaLop

WHERE MaSV = @MaSV";

sqlCmd.Connection = sqlCon;

// Thêm các parameter vào lệnh SQL

sqlCmd.Parameters.AddWithValue("@MaSV", maSV);

sqlCmd.Parameters.AddWithValue("@TenSV", tenSV);

sqlCmd.Parameters.AddWithValue("@GioiTinh", gioiTinh);

sqlCmd.Parameters.AddWithValue("@NgaySinh", ngaySinh);

sqlCmd.Parameters.AddWithValue("@QueQuan", queQuan);

sqlCmd.Parameters.AddWithValue("@MaLop", maLop);

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("Cập nhật thông tin sinh viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiDSinhVienTheoLop(maLop);

}

else

{

MessageBox.Show("Không có bản ghi nào được cập nhật!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

// Xóa dữ liệu trên form

txtMaSV.Clear();

txtTenSV.Clear();

txtQueQuan.Clear();

txtMaLop.Clear();

cbGioiTinh.SelectedIndex = -1;

}

catch (Exception ex)

{

MessageBox.Show("Lỗi cập nhật: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

finally

{

DongKetNoi();

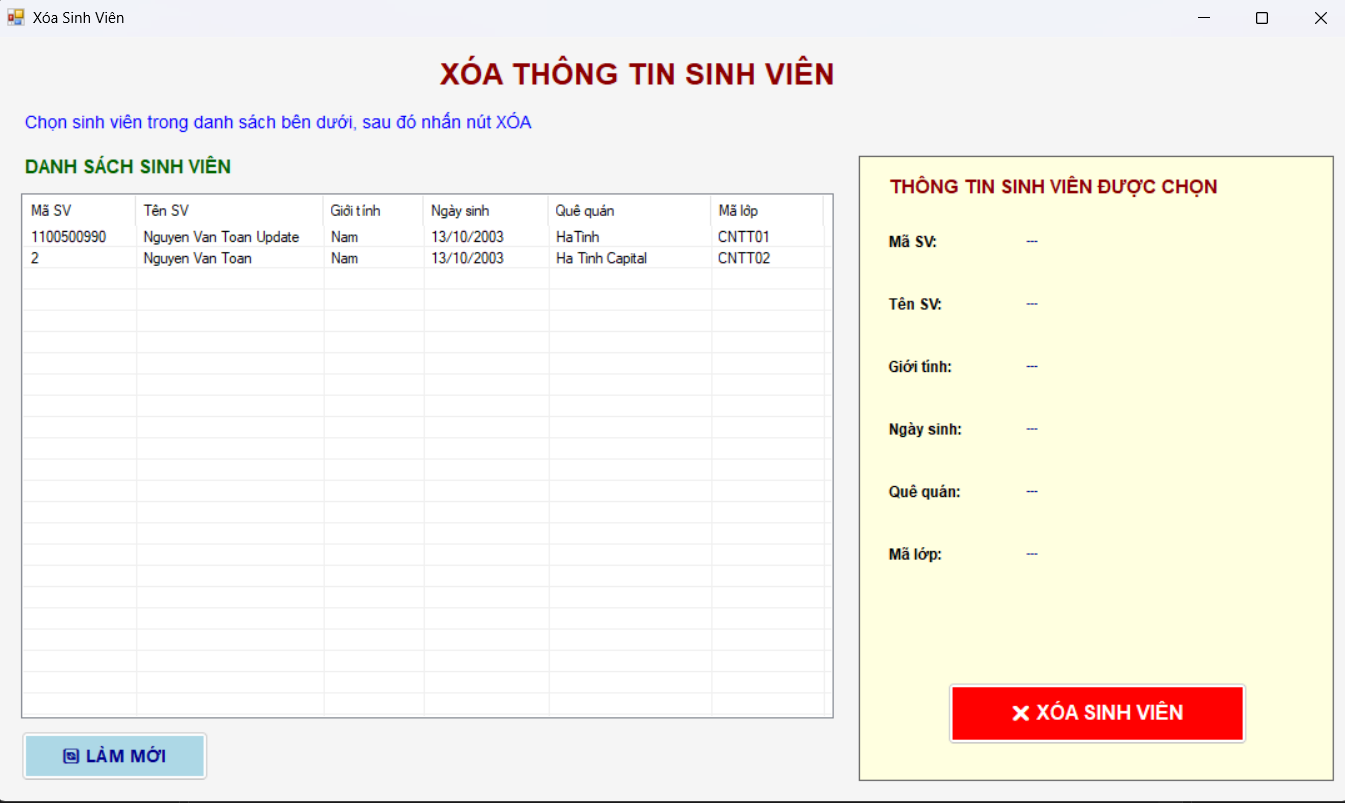
}

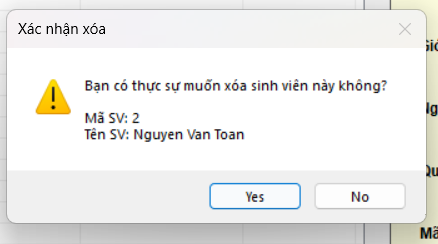
}

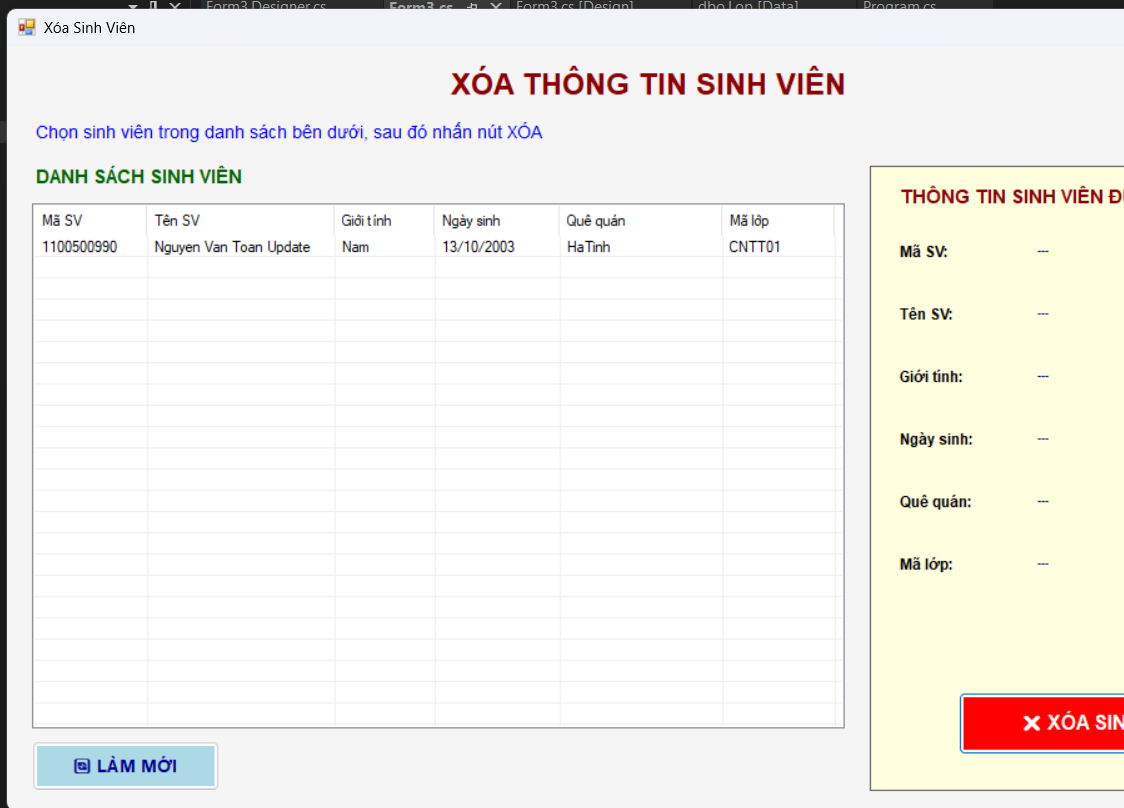
}

}

# Thực hành 4

****

****

****

## Code:

using System;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Windows.Forms;

namespace \_11500800040\_NguyenVanToan\_Lab5

{

public partial class Form3 : Form

{

// Chuỗi kết nối

string strCon = @"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\VanToan123\source\repos\11500800040\_NguyenVanToan\_Lab5\11500800040\_NguyenVanToan\_Lab5\Lab5DB.mdf;Integrated Security=True";

// Đối tượng kết nối

SqlConnection sqlCon = null;

// Khai báo controls

private ListView lsvDanhSach;

private Button btnXoaSV;

private Button btnLamMoi;

private Label lblTitle;

private Label lblDanhSach;

private Label lblHuongDan;

private Panel panelThongTin;

private Label lblMaSV;

private Label lblTenSV;

private Label lblGioiTinh;

private Label lblNgaySinh;

private Label lblQueQuan;

private Label lblMaLop;

private Label lblValueMaSV;

private Label lblValueTenSV;

private Label lblValueGioiTinh;

private Label lblValueNgaySinh;

private Label lblValueQueQuan;

private Label lblValueMaLop;

// Biến lưu mã sinh viên được chọn

string maSV = "";

public Form3()

{

InitializeComponent();

}

private void InitializeComponent()

{

this.Text = "Xóa Sinh Viên";

this.Size = new Size(1100, 650);

this.StartPosition = FormStartPosition.CenterScreen;

this.BackColor = Color.WhiteSmoke;

// Title

lblTitle = new Label();

lblTitle.Text = "XÓA THÔNG TIN SINH VIÊN";

lblTitle.Font = new Font("Arial", 18, FontStyle.Bold);

lblTitle.ForeColor = Color.DarkRed;

lblTitle.Location = new Point(350, 15);

lblTitle.Size = new Size(450, 35);

this.Controls.Add(lblTitle);

// Hướng dẫn

lblHuongDan = new Label();

lblHuongDan.Text = "Chọn sinh viên trong danh sách bên dưới, sau đó nhấn nút XÓA";

lblHuongDan.Font = new Font("Arial", 10);

lblHuongDan.ForeColor = Color.Blue;

lblHuongDan.Location = new Point(20, 60);

lblHuongDan.Size = new Size(650, 25);

this.Controls.Add(lblHuongDan);

// Label Danh sách

lblDanhSach = new Label();

lblDanhSach.Text = "DANH SÁCH SINH VIÊN";

lblDanhSach.Location = new Point(20, 95);

lblDanhSach.Size = new Size(300, 25);

lblDanhSach.Font = new Font("Arial", 11, FontStyle.Bold);

lblDanhSach.ForeColor = Color.DarkGreen;

this.Controls.Add(lblDanhSach);

// ListView Danh sách sinh viên

lsvDanhSach = new ListView();

lsvDanhSach.Location = new Point(20, 125);

lsvDanhSach.Size = new Size(650, 420);

lsvDanhSach.View = View.Details;

lsvDanhSach.FullRowSelect = true;

lsvDanhSach.GridLines = true;

lsvDanhSach.BackColor = Color.White;

lsvDanhSach.Columns.Add("Mã SV", 90);

lsvDanhSach.Columns.Add("Tên SV", 150);

lsvDanhSach.Columns.Add("Giới tính", 80);

lsvDanhSach.Columns.Add("Ngày sinh", 100);

lsvDanhSach.Columns.Add("Quê quán", 130);

lsvDanhSach.Columns.Add("Mã lớp", 90);

lsvDanhSach.SelectedIndexChanged += new EventHandler(lsvDanhSach\_SelectedIndexChanged);

this.Controls.Add(lsvDanhSach);

// Button Làm mới

btnLamMoi = new Button();

btnLamMoi.Text = "🔄 LÀM MỚI";

btnLamMoi.Location = new Point(20, 555);

btnLamMoi.Size = new Size(150, 40);

btnLamMoi.BackColor = Color.LightBlue;

btnLamMoi.Font = new Font("Arial", 10, FontStyle.Bold);

btnLamMoi.ForeColor = Color.DarkBlue;

btnLamMoi.Cursor = Cursors.Hand;

btnLamMoi.Click += new EventHandler(btnLamMoi\_Click);

this.Controls.Add(btnLamMoi);

// Panel Thông tin sinh viên được chọn

panelThongTin = new Panel();

panelThongTin.Location = new Point(690, 95);

panelThongTin.Size = new Size(380, 500);

panelThongTin.BorderStyle = BorderStyle.FixedSingle;

panelThongTin.BackColor = Color.LightYellow;

// Title Panel

Label lblPanelTitle = new Label();

lblPanelTitle.Text = "THÔNG TIN SINH VIÊN ĐƯỢC CHỌN";

lblPanelTitle.Font = new Font("Arial", 11, FontStyle.Bold);

lblPanelTitle.ForeColor = Color.DarkRed;

lblPanelTitle.Location = new Point(20, 15);

lblPanelTitle.Size = new Size(340, 25);

panelThongTin.Controls.Add(lblPanelTitle);

// Labels trong panel

int startY = 60;

int spacing = 50;

lblMaSV = new Label() { Text = "Mã SV:", Location = new Point(20, startY), Size = new Size(100, 20), Font = new Font("Arial", 9, FontStyle.Bold) };

lblValueMaSV = new Label() { Text = "---", Location = new Point(130, startY), Size = new Size(230, 20), ForeColor = Color.DarkBlue };

lblTenSV = new Label() { Text = "Tên SV:", Location = new Point(20, startY + spacing), Size = new Size(100, 20), Font = new Font("Arial", 9, FontStyle.Bold) };

lblValueTenSV = new Label() { Text = "---", Location = new Point(130, startY + spacing), Size = new Size(230, 20), ForeColor = Color.DarkBlue };

lblGioiTinh = new Label() { Text = "Giới tính:", Location = new Point(20, startY + spacing \* 2), Size = new Size(100, 20), Font = new Font("Arial", 9, FontStyle.Bold) };

lblValueGioiTinh = new Label() { Text = "---", Location = new Point(130, startY + spacing \* 2), Size = new Size(230, 20), ForeColor = Color.DarkBlue };

lblNgaySinh = new Label() { Text = "Ngày sinh:", Location = new Point(20, startY + spacing \* 3), Size = new Size(100, 20), Font = new Font("Arial", 9, FontStyle.Bold) };

lblValueNgaySinh = new Label() { Text = "---", Location = new Point(130, startY + spacing \* 3), Size = new Size(230, 20), ForeColor = Color.DarkBlue };

lblQueQuan = new Label() { Text = "Quê quán:", Location = new Point(20, startY + spacing \* 4), Size = new Size(100, 20), Font = new Font("Arial", 9, FontStyle.Bold) };

lblValueQueQuan = new Label() { Text = "---", Location = new Point(130, startY + spacing \* 4), Size = new Size(230, 20), ForeColor = Color.DarkBlue };

lblMaLop = new Label() { Text = "Mã lớp:", Location = new Point(20, startY + spacing \* 5), Size = new Size(100, 20), Font = new Font("Arial", 9, FontStyle.Bold) };

lblValueMaLop = new Label() { Text = "---", Location = new Point(130, startY + spacing \* 5), Size = new Size(230, 20), ForeColor = Color.DarkBlue };

panelThongTin.Controls.Add(lblMaSV);

panelThongTin.Controls.Add(lblValueMaSV);

panelThongTin.Controls.Add(lblTenSV);

panelThongTin.Controls.Add(lblValueTenSV);

panelThongTin.Controls.Add(lblGioiTinh);

panelThongTin.Controls.Add(lblValueGioiTinh);

panelThongTin.Controls.Add(lblNgaySinh);

panelThongTin.Controls.Add(lblValueNgaySinh);

panelThongTin.Controls.Add(lblQueQuan);

panelThongTin.Controls.Add(lblValueQueQuan);

panelThongTin.Controls.Add(lblMaLop);

panelThongTin.Controls.Add(lblValueMaLop);

// Button Xóa trong panel

btnXoaSV = new Button();

btnXoaSV.Text = " XÓA SINH VIÊN";

btnXoaSV.Location = new Point(70, 420);

btnXoaSV.Size = new Size(240, 50);

btnXoaSV.BackColor = Color.Red;

btnXoaSV.Font = new Font("Arial", 12, FontStyle.Bold);

btnXoaSV.ForeColor = Color.White;

btnXoaSV.Cursor = Cursors.Hand;

btnXoaSV.Click += new EventHandler(btnXoaSV\_Click);

panelThongTin.Controls.Add(btnXoaSV);

this.Controls.Add(panelThongTin);

// Load event

this.Load += new EventHandler(Form3\_Load);

}

// Hàm mở kết nối

private void MoKetNoi()

{

if (sqlCon == null)

{

sqlCon = new SqlConnection(strCon);

}

if (sqlCon.State == ConnectionState.Closed)

{

sqlCon.Open();

}

}

// Hàm đóng kết nối

private void DongKetNoi()

{

if (sqlCon != null && sqlCon.State == ConnectionState.Open)

{

sqlCon.Close();

}

}

// Hàm hiển thị danh sách sinh viên

private void HienThiDSSinhVien()

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "select \* from SinhVien";

sqlCmd.Connection = sqlCon;

lsvDanhSach.Items.Clear();

SqlDataReader reader = sqlCmd.ExecuteReader();

while (reader.Read())

{

string maSV = reader.GetString(0);

string tenSV = reader.GetString(1);

string gioiTinh = reader.GetString(2);

string ngaySinh = reader.GetDateTime(3).ToString("dd/MM/yyyy");

string queQuan = reader.GetString(4);

string maLop = reader.GetString(5);

ListViewItem lvi = new ListViewItem(maSV);

lvi.SubItems.Add(tenSV);

lvi.SubItems.Add(gioiTinh);

lvi.SubItems.Add(ngaySinh);

lvi.SubItems.Add(queQuan);

lvi.SubItems.Add(maLop);

lsvDanhSach.Items.Add(lvi);

}

reader.Close();

}

catch (Exception ex)

{

MessageBox.Show("Lỗi hiển thị dữ liệu: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

// Sự kiện Load form

private void Form3\_Load(object sender, EventArgs e)

{

HienThiDSSinhVien();

}

// Sự kiện khi chọn sinh viên trong ListView

private void lsvDanhSach\_SelectedIndexChanged(object sender, EventArgs e)

{

// Nếu chưa chọn sinh viên nào để xóa

if (lsvDanhSach.SelectedItems.Count == 0)

{

maSV = "";

// Reset thông tin hiển thị

lblValueMaSV.Text = "---";

lblValueTenSV.Text = "---";

lblValueGioiTinh.Text = "---";

lblValueNgaySinh.Text = "---";

lblValueQueQuan.Text = "---";

lblValueMaLop.Text = "---";

return;

}

// Nếu đã chọn 1 sinh viên

ListViewItem lvi = lsvDanhSach.SelectedItems[0];

maSV = lvi.SubItems[0].Text.Trim();

// Hiển thị thông tin sinh viên được chọn

lblValueMaSV.Text = lvi.SubItems[0].Text;

lblValueTenSV.Text = lvi.SubItems[1].Text;

lblValueGioiTinh.Text = lvi.SubItems[2].Text;

lblValueNgaySinh.Text = lvi.SubItems[3].Text;

lblValueQueQuan.Text = lvi.SubItems[4].Text;

lblValueMaLop.Text = lvi.SubItems[5].Text;

}

// Sự kiện nút Xóa

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

{

if (maSV == "")

{

MessageBox.Show("Bạn chưa chọn sinh viên nào để xóa!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

DialogResult result = MessageBox.Show(

"Bạn có thực sự muốn xóa sinh viên này không?\n\n" +

"Mã SV: " + lblValueMaSV.Text + "\n" +

"Tên SV: " + lblValueTenSV.Text,

"Xác nhận xóa",

MessageBoxButtons.YesNo,

MessageBoxIcon.Warning);

if (result == DialogResult.Yes)

{

XoaSV(maSV);

}

}

}

// Hàm xóa sinh viên

private void XoaSV(string maSV)

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "delete from SinhVien where MaSV='" + maSV + "'";

sqlCmd.Connection = sqlCon;

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("Xóa sinh viên thành công!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

// Reset thông tin và làm mới danh sách

this.maSV = "";

lblValueMaSV.Text = "---";

lblValueTenSV.Text = "---";

lblValueGioiTinh.Text = "---";

lblValueNgaySinh.Text = "---";

lblValueQueQuan.Text = "---";

lblValueMaLop.Text = "---";

HienThiDSSinhVien();

}

else

{

MessageBox.Show("Xóa sinh viên không thành công!", "Lỗi",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

catch (Exception ex)

{

MessageBox.Show("Lỗi xóa dữ liệu: " + ex.Message, "Lỗi",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

// Sự kiện nút Làm mới

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

{

HienThiDSSinhVien();

maSV = "";

lblValueMaSV.Text = "---";

lblValueTenSV.Text = "---";

lblValueGioiTinh.Text = "---";

lblValueNgaySinh.Text = "---";

lblValueQueQuan.Text = "---";

lblValueMaLop.Text = "---";

MessageBox.Show("Đã làm mới danh sách!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

}

## Áp dụng 3

Hàm xóa sinh viên ko có sử dụng parameter  
private void XoaSV(string maSV)

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "delete from SinhVien where MaSV='" + maSV + "'";

sqlCmd.Connection = sqlCon;

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("Xóa sinh viên thành công!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

// Reset thông tin và làm mới danh sách

this.maSV = "";

lblValueMaSV.Text = "---";

lblValueTenSV.Text = "---";

lblValueGioiTinh.Text = "---";

lblValueNgaySinh.Text = "---";

lblValueQueQuan.Text = "---";

lblValueMaLop.Text = "---";

HienThiDSSinhVien();

}

else

{

MessageBox.Show("Xóa sinh viên không thành công!", "Lỗi",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

catch (Exception ex)

{

MessageBox.Show("Lỗi xóa dữ liệu: " + ex.Message, "Lỗi",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}  
  
--------------------------------------

Còn đây là hàm xoaSV() sử dụng parameter:

private void XoaSV(string maSV)

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "DELETE FROM SinhVien WHERE MaSV = @MaSV";

sqlCmd.Connection = sqlCon;

sqlCmd.Parameters.AddWithValue("@MaSV", maSV);

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("✅ Xóa sinh viên thành công!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

this.maSV = "";

lblValueMaSV.Text = "---";

lblValueTenSV.Text = "---";

lblValueGioiTinh.Text = "---";

lblValueNgaySinh.Text = "---";

lblValueQueQuan.Text = "---";

lblValueMaLop.Text = "---";

HienThiDSSinhVien();

}

else

{

MessageBox.Show("Không có sinh viên nào bị xóa!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

catch (Exception ex)

{

MessageBox.Show("❌ Lỗi khi xóa dữ liệu: " + ex.Message, "Lỗi",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

finally

{

DongKetNoi();

}

}