**BÀI TẬP BUỔI 6**

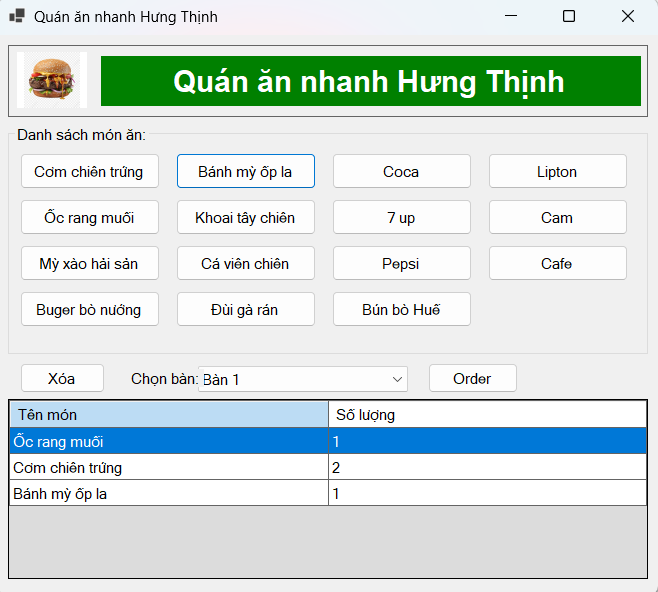
**Lab3:**

**Thực hành 1:**

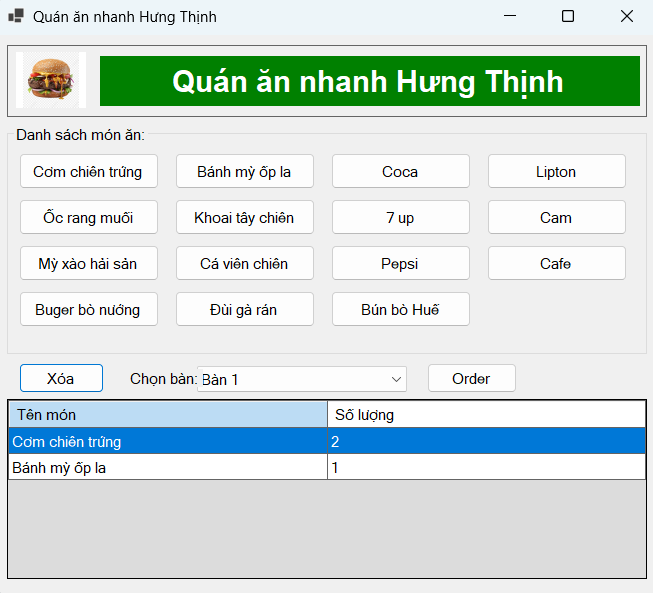
Tạo project trên ThucHanh1Lab3.cs



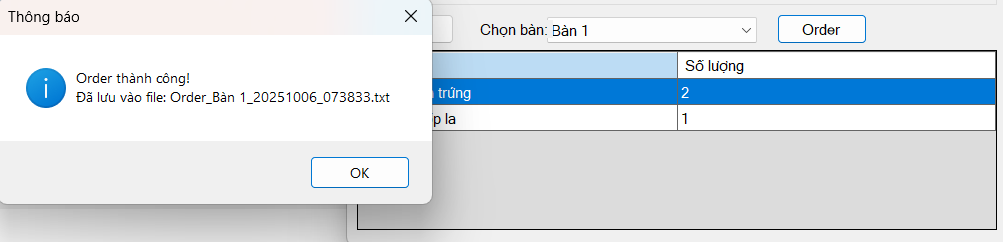
Chọn món:



Xóa món:



Order món:



Code trong ThucHanh1Lab3.cs:

using System;

using System.Data;

using System.Drawing;

using System.IO;

using System.Windows.Forms;

namespace \_1150080022\_buoi6\_LeTuanKiet

{

    public partial class Form1 : Form

    {

        private DataTable dtOrder;

        private ComboBox cmbBan;

        private DataGridView dgvOrder;

        private Button btnOrder, btnXoa;

        public Form1()

        {

            InitializeComponent();

            KhoiTaoGiaoDien();

            KhoiTaoDataTable();

        }

        private void KhoiTaoGiaoDien()

        {

            // Form rộng + cao hơn một chút

            Text = "Quán ăn nhanh Hưng Thịnh";

            StartPosition = FormStartPosition.CenterScreen;

            ClientSize = new Size(660, 560);

            Font = new Font("Microsoft Sans Serif", 9F);

            // HEADER

            var pnlHeader = new Panel

            {

                Location = new Point(10, 10),

                Size = new Size(640, 72),

                BorderStyle = BorderStyle.FixedSingle

            };

            var picLogo = new PictureBox

            {

                Location = new Point(8, 7),

                Size = new Size(70, 56),

                SizeMode = PictureBoxSizeMode.Zoom,

                BackColor = Color.White

            };

            picLogo.Load("https://img.lovepik.com/png/20231106/gourmet-burger-cheese-Hamburger-cheeseburger-A-meal\_506563\_wh860.png");

            var lblBanner = new Label

            {

                Text = "Quán ăn nhanh Hưng Thịnh",

                Location = new Point(92, 10),

                Size = new Size(540, 50),

                BackColor = Color.Green,

                ForeColor = Color.White,

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

                TextAlign = ContentAlignment.MiddleCenter

            };

            pnlHeader.Controls.Add(picLogo);

            pnlHeader.Controls.Add(lblBanner);

            Controls.Add(pnlHeader);

            // DANH SÁCH MÓN ĂN (cao + rộng hơn)

            var grp = new GroupBox

            {

                Text = "Danh sách món ăn:",

                Location = new Point(10, 90),

                Size = new Size(640, 230) // tăng chiều cao

            };

            Controls.Add(grp);

            // helper tạo nút món – tăng chút kích thước & khoảng cách

            Button TaoButton(string text, int col, int row)

            {

                var btn = new Button

                {

                    Text = text,

                    Size = new Size(140, 36),

                    Location = new Point(12 + col \* 156, 28 + row \* 46)

                };

                btn.Click += BtnMonAn\_Click;

                return btn;

            }

            // Hàng 1

            grp.Controls.Add(TaoButton("Cơm chiên trứng", 0, 0));

            grp.Controls.Add(TaoButton("Bánh mỳ ốp la", 1, 0));

            grp.Controls.Add(TaoButton("Coca", 2, 0));

            grp.Controls.Add(TaoButton("Lipton", 3, 0));

            // Hàng 2

            grp.Controls.Add(TaoButton("Ốc rang muối", 0, 1));

            grp.Controls.Add(TaoButton("Khoai tây chiên", 1, 1));

            grp.Controls.Add(TaoButton("7 up", 2, 1));

            grp.Controls.Add(TaoButton("Cam", 3, 1));

            // Hàng 3

            grp.Controls.Add(TaoButton("Mỳ xào hải sản", 0, 2));

            grp.Controls.Add(TaoButton("Cá viên chiên", 1, 2));

            grp.Controls.Add(TaoButton("Pepsi", 2, 2));

            grp.Controls.Add(TaoButton("Cafe", 3, 2));

            // Hàng 4

            grp.Controls.Add(TaoButton("Buger bò nướng", 0, 3));

            grp.Controls.Add(TaoButton("Đùi gà rán", 1, 3));

            grp.Controls.Add(TaoButton("Bún bò Huế", 2, 3));

            // DÒNG CHỨC NĂNG (dưới groupbox)

            int yFn = grp.Bottom + 8;

            btnXoa = new Button { Text = "Xóa", Location = new Point(22, yFn), Size = new Size(85, 30) };

            btnXoa.Click += BtnXoa\_Click;

            Controls.Add(btnXoa);

            var lblChonBan = new Label { Text = "Chọn bàn:", Location = new Point(130, yFn + 6), AutoSize = true };

            Controls.Add(lblChonBan);

            cmbBan = new ComboBox

            {

                Location = new Point(200, yFn + 3),

                Size = new Size(210, 25),

                DropDownStyle = ComboBoxStyle.DropDownList

            };

            for (int i = 1; i <= 10; i++) cmbBan.Items.Add($"Bàn {i}");

            cmbBan.SelectedIndex = 0;

            Controls.Add(cmbBan);

            btnOrder = new Button { Text = "Order", Location = new Point(430, yFn), Size = new Size(90, 30) };

            btnOrder.Click += BtnOrder\_Click;

            Controls.Add(btnOrder);

            // LƯỚI ORDER (cao hơn)

            dgvOrder = new DataGridView

            {

                Location = new Point(10, yFn + 36),

                Size = new Size(640, 180), // tăng chiều cao

                AutoSizeColumnsMode = DataGridViewAutoSizeColumnsMode.Fill,

                AllowUserToAddRows = false,

                ReadOnly = true,

                SelectionMode = DataGridViewSelectionMode.FullRowSelect,

                BackgroundColor = Color.Gainsboro,

                RowHeadersVisible = false

            };

            dgvOrder.RowTemplate.Height = 26;

            dgvOrder.ColumnHeadersHeight = 28;

            Controls.Add(dgvOrder);

        }

        private void KhoiTaoDataTable()

        {

            dtOrder = new DataTable();

            dtOrder.Columns.Add("Tên món", typeof(string));

            dtOrder.Columns.Add("Số lượng", typeof(int));

            dgvOrder.DataSource = dtOrder;

        }

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

        {

            var tenMon = ((Button)sender).Text;

            var rows = dtOrder.Select($"[Tên món] = '{tenMon.Replace("'", "''")}'");

            if (rows.Length > 0) rows[0]["Số lượng"] = (int)rows[0]["Số lượng"] + 1;

            else dtOrder.Rows.Add(tenMon, 1);

        }

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

        {

            if (dgvOrder.SelectedRows.Count == 0)

            {

                MessageBox.Show("Vui lòng chọn món cần xóa!", "Thông báo",

                    MessageBoxButtons.OK, MessageBoxIcon.Warning);

                return;

            }

            dtOrder.Rows[dgvOrder.SelectedRows[0].Index].Delete();

        }

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

        {

            if (dtOrder.Rows.Count == 0)

            {

                MessageBox.Show("Chưa có món nào được chọn!", "Thông báo",

                    MessageBoxButtons.OK, MessageBoxIcon.Warning);

                return;

            }

            string tenBan = cmbBan.SelectedItem.ToString();

            string fileName = $"Order\_{tenBan}\_{DateTime.Now:yyyyMMdd\_HHmmss}.txt";

            try

            {

                using (var sw = new StreamWriter(fileName))

                {

                    sw.WriteLine("=====================================");

                    sw.WriteLine("     QUÁN ĂN NHANH HƯNG THỊNH");

                    sw.WriteLine("=====================================");

                    sw.WriteLine($"Bàn: {tenBan}");

                    sw.WriteLine($"Thời gian: {DateTime.Now:dd/MM/yyyy HH:mm:ss}");

                    sw.WriteLine("-------------------------------------");

                    sw.WriteLine("TÊN MÓN\t\t\tSỐ LƯỢNG");

                    sw.WriteLine("-------------------------------------");

                    foreach (DataRow row in dtOrder.Rows)

                        sw.WriteLine($"{row["Tên món"]}\t\t{row["Số lượng"]}");

                    sw.WriteLine("=====================================");

                }

                MessageBox.Show($"Order thành công!\nĐã lưu vào file: {fileName}",

                    "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

                dtOrder.Clear();

            }

            catch (Exception ex)

            {

                MessageBox.Show($"Lỗi khi ghi file: {ex.Message}", "Lỗi",

                    MessageBoxButtons.OK, MessageBoxIcon.Error);

            }

        }

        private void InitializeComponent() { }

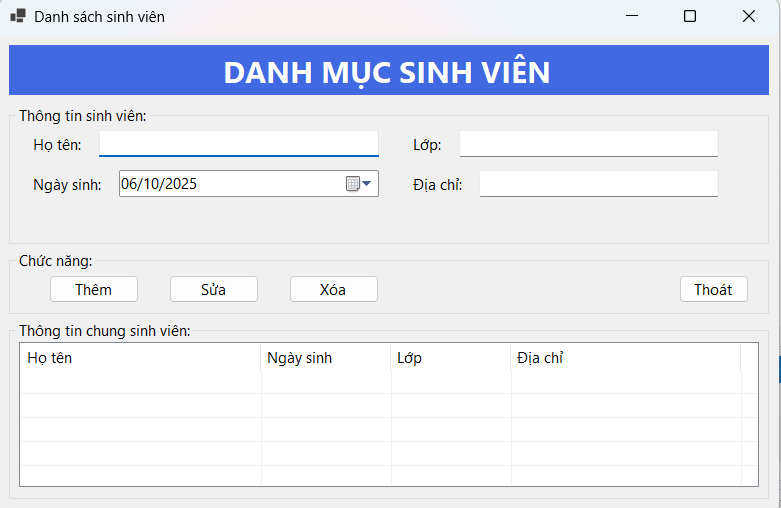
    }

}

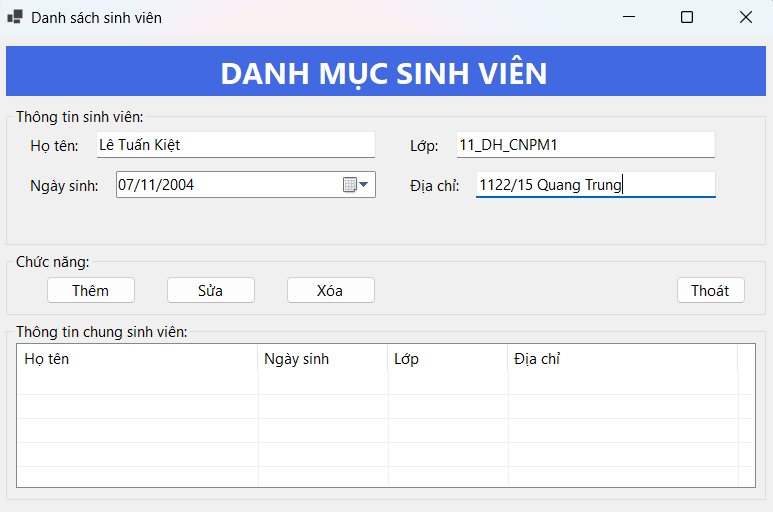
Program.cs là Form1

**Thực hành 2:**

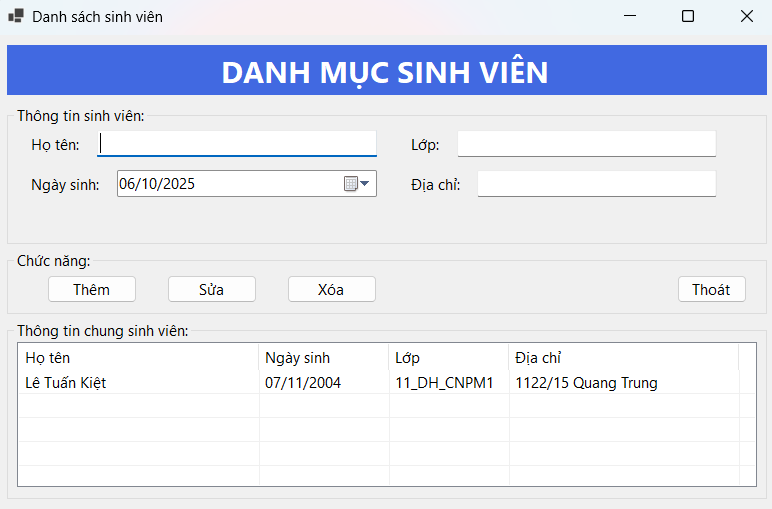
Tạo project ThucHanh2Lab3.cs



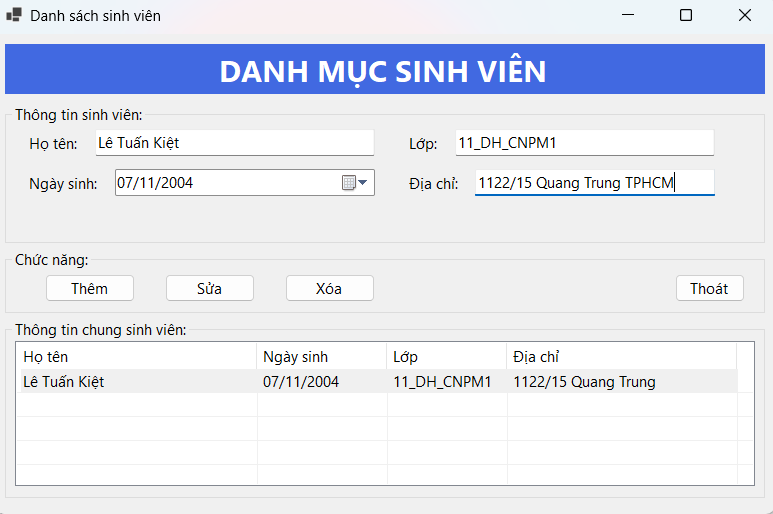
Điền thông tin:



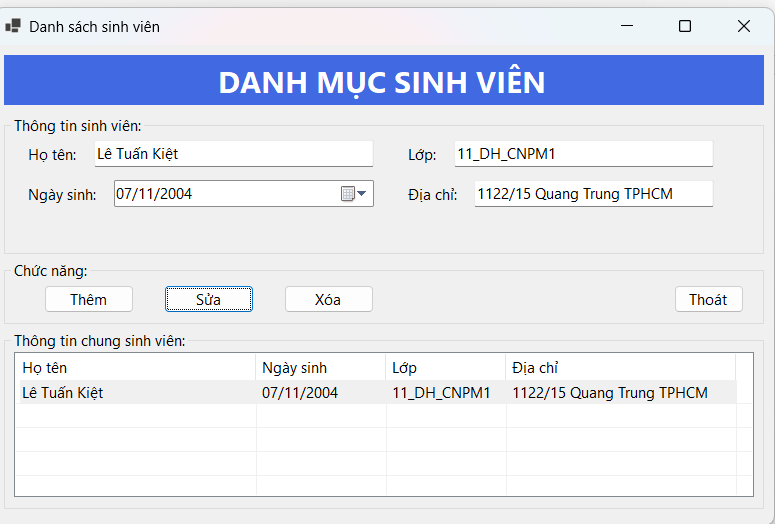
Thêm sinh viên:



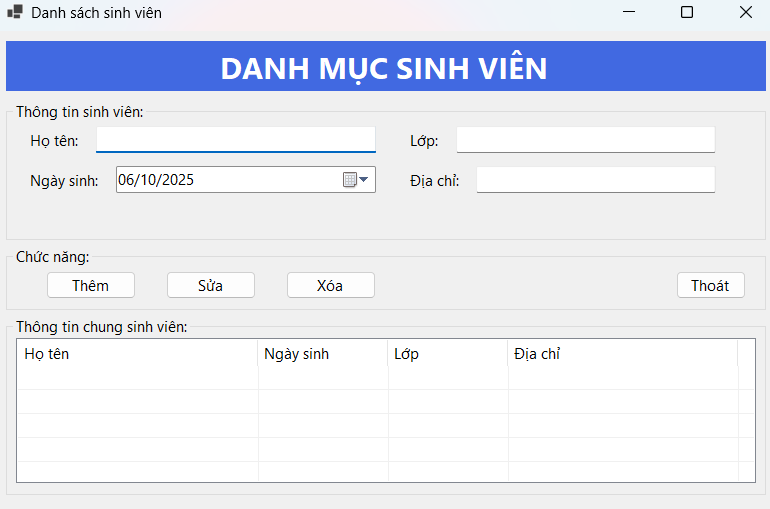
Sửa thông tin:



Cập nhật thông tin:



Xóa:



Code trong ThucHanh2Lab3.cs:

using System;

using System.Drawing;

using System.Windows.Forms;

namespace \_1150080022\_buoi6\_LeTuanKiet

{

    public partial class Form2 : Form

    {

        TextBox txtHoTen, txtLop, txtDiaChi;

        DateTimePicker dtpNgaySinh;

        Button btnThem, btnSua, btnXoa, btnThoat;

        ListView lvSV;

        public Form2()

        {

            InitializeComponent();

            TaoGiaoDien();

        }

        private void TaoGiaoDien()

        {

            Text = "Danh sách sinh viên";

            StartPosition = FormStartPosition.CenterScreen;

            ClientSize = new Size(780, 480);

            Font = new Font("Segoe UI", 9F);

            // Title

            var lblTitle = new Label

            {

                Text = "DANH MỤC SINH VIÊN",

                Font = new Font("Segoe UI", 18, FontStyle.Bold),

                ForeColor = Color.White,

                BackColor = Color.RoyalBlue,

                TextAlign = ContentAlignment.MiddleCenter,

                Location = new Point(10, 10),

                Size = new Size(760, 50)

            };

            Controls.Add(lblTitle);

            // Group thông tin

            var grpInfo = new GroupBox

            {

                Text = "Thông tin sinh viên:",

                Location = new Point(10, 70),

                Size = new Size(760, 140)

            };

            Controls.Add(grpInfo);

            int left1 = 20, left2 = 400, top = 25, h = 26, w1 = 280, w2 = 320;

            grpInfo.Controls.Add(new Label { Text = "Họ tên:", Location = new Point(left1, top + 4), AutoSize = true });

            txtHoTen = new TextBox { Location = new Point(left1 + 70, top), Size = new Size(w1, h) };

            grpInfo.Controls.Add(txtHoTen);

            grpInfo.Controls.Add(new Label { Text = "Lớp:", Location = new Point(left2, top + 4), AutoSize = true });

            txtLop = new TextBox { Location = new Point(left2 + 50, top), Size = new Size(w2 - 60, h) };

            grpInfo.Controls.Add(txtLop);

            grpInfo.Controls.Add(new Label { Text = "Ngày sinh:", Location = new Point(left1, top + 44), AutoSize = true });

            dtpNgaySinh = new DateTimePicker

            {

                Location = new Point(left1 + 90, top + 40),

                Size = new Size(w1 - 20, h),

                Format = DateTimePickerFormat.Custom,

                CustomFormat = "dd/MM/yyyy"

            };

            grpInfo.Controls.Add(dtpNgaySinh);

            grpInfo.Controls.Add(new Label { Text = "Địa chỉ:", Location = new Point(left2, top + 44), AutoSize = true });

            txtDiaChi = new TextBox { Location = new Point(left2 + 70, top + 40), Size = new Size(w2 - 80, h) };

            grpInfo.Controls.Add(txtDiaChi);

            // Chức năng

            var grpActions = new GroupBox

            {

                Text = "Chức năng:",

                Location = new Point(10, 215),

                Size = new Size(760, 65)

            };

            Controls.Add(grpActions);

            btnThem = new Button { Text = "Thêm", Location = new Point(40, 25), Size = new Size(90, 28) };

            btnSua = new Button { Text = "Sửa", Location = new Point(160, 25), Size = new Size(90, 28) };

            btnXoa = new Button { Text = "Xóa", Location = new Point(280, 25), Size = new Size(90, 28) };

            btnThoat = new Button { Text = "Thoát", Location = new Point(670, 25), Size = new Size(70, 28) };

            btnThem.Click += BtnThem\_Click;

            btnSua.Click += BtnSua\_Click;

            btnXoa.Click += BtnXoa\_Click;

            btnThoat.Click += (s, e) => Close();

            grpActions.Controls.AddRange(new Control[] { btnThem, btnSua, btnXoa, btnThoat });

            // Listview

            var grpList = new GroupBox

            {

                Text = "Thông tin chung sinh viên:",

                Location = new Point(10, 285),

                Size = new Size(760, 180)

            };

            Controls.Add(grpList);

            lvSV = new ListView

            {

                Location = new Point(10, 22),

                Size = new Size(740, 145),

                View = View.Details,

                FullRowSelect = true,

                GridLines = true

            };

            lvSV.Columns.Add("Họ tên", 240);

            lvSV.Columns.Add("Ngày sinh", 130);

            lvSV.Columns.Add("Lớp", 120);

            lvSV.Columns.Add("Địa chỉ", 230);

            lvSV.SelectedIndexChanged += LvSV\_SelectedIndexChanged;

            grpList.Controls.Add(lvSV);

        }

        // ====== HANDLERS ======

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

        {

            if (string.IsNullOrWhiteSpace(txtHoTen.Text))

            {

                MessageBox.Show("Họ tên không được rỗng!", "Thông báo",

                    MessageBoxButtons.OK, MessageBoxIcon.Warning);

                txtHoTen.Focus();

                return;

            }

            var item = new ListViewItem(txtHoTen.Text.Trim());

            item.SubItems.Add(dtpNgaySinh.Value.ToString("dd/MM/yyyy"));

            item.SubItems.Add(txtLop.Text.Trim());

            item.SubItems.Add(txtDiaChi.Text.Trim());

            lvSV.Items.Add(item);

            ClearInputs();

        }

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

        {

            if (lvSV.SelectedItems.Count == 0)

            {

                MessageBox.Show("Vui lòng chọn 1 dòng để xóa!", "Thông báo",

                    MessageBoxButtons.OK, MessageBoxIcon.Warning);

                return;

            }

            lvSV.Items.Remove(lvSV.SelectedItems[0]);

            ClearInputs();

        }

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

        {

            if (lvSV.SelectedItems.Count == 0)

            {

                MessageBox.Show("Vui lòng chọn 1 dòng để sửa!", "Thông báo",

                    MessageBoxButtons.OK, MessageBoxIcon.Warning);

                return;

            }

            if (string.IsNullOrWhiteSpace(txtHoTen.Text))

            {

                MessageBox.Show("Họ tên không được rỗng!", "Thông báo",

                    MessageBoxButtons.OK, MessageBoxIcon.Warning);

                txtHoTen.Focus();

                return;

            }

            var it = lvSV.SelectedItems[0];

            it.Text = txtHoTen.Text.Trim();

            it.SubItems[1].Text = dtpNgaySinh.Value.ToString("dd/MM/yyyy");

            it.SubItems[2].Text = txtLop.Text.Trim();

            it.SubItems[3].Text = txtDiaChi.Text.Trim();

        }

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

        {

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

            var it = lvSV.SelectedItems[0];

            txtHoTen.Text = it.Text;

            DateTime d;

            if (DateTime.TryParseExact(it.SubItems[1].Text, "dd/MM/yyyy",

                System.Globalization.CultureInfo.InvariantCulture,

                System.Globalization.DateTimeStyles.None, out d))

            {

                dtpNgaySinh.Value = d;

            }

            txtLop.Text = it.SubItems[2].Text;

            txtDiaChi.Text = it.SubItems[3].Text;

        }

        // ====== UTIL ======

        private void ClearInputs()

        {

            txtHoTen.Clear();

            txtLop.Clear();

            txtDiaChi.Clear();

            dtpNgaySinh.Value = DateTime.Today;

            txtHoTen.Focus();

        }

        // Designer stub

        private void InitializeComponent() { }

    }

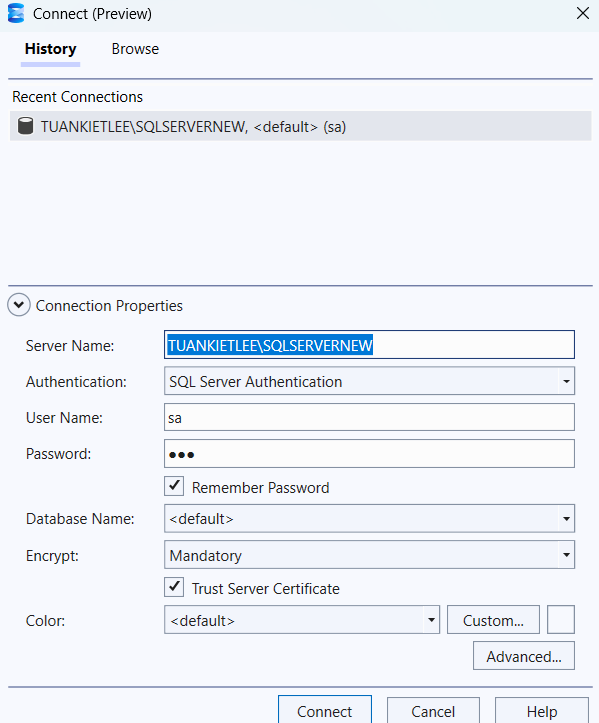
}

Program.cs là Form2

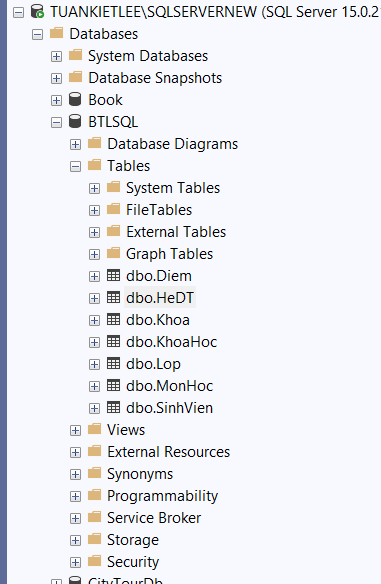
**LAB 4**

**Thực hành 1: Tạo kết nối giữa Winform và CSDL.**

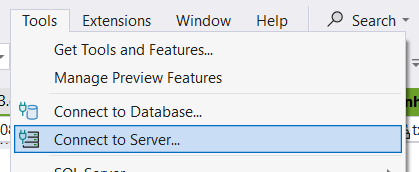
Đầu tiên, ta phải có được Server name của máy local mình đang sử dụng:

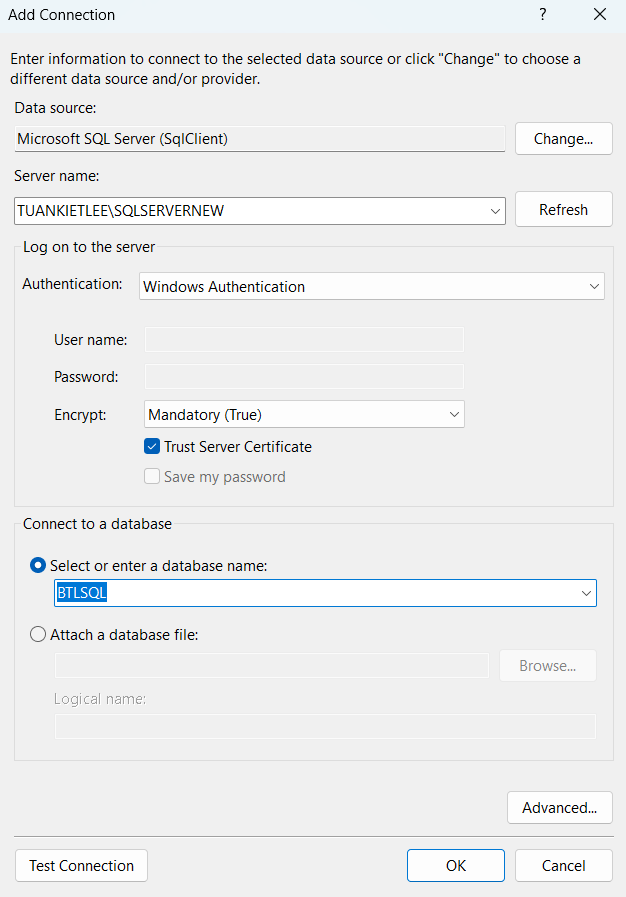


Tạo CSDL:

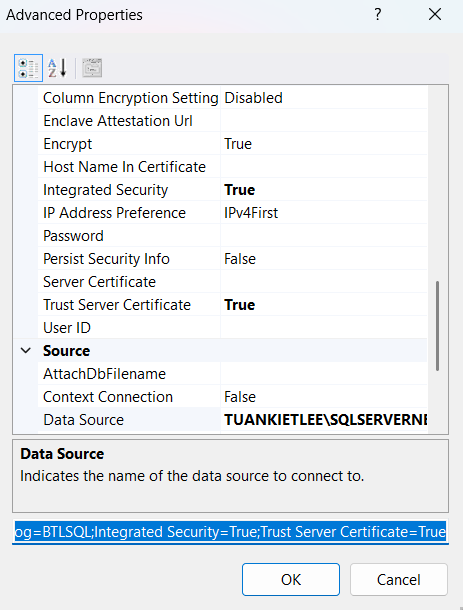


Tiếp theo, trong VS, check trong TOOLS/Connect to database...

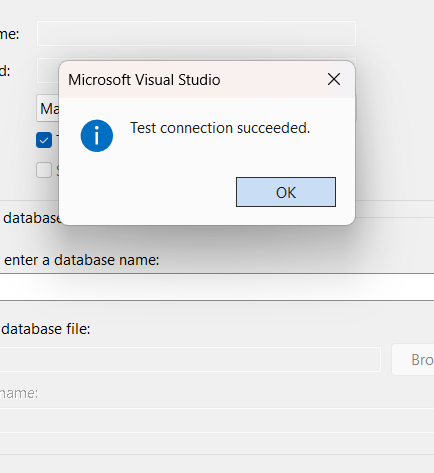




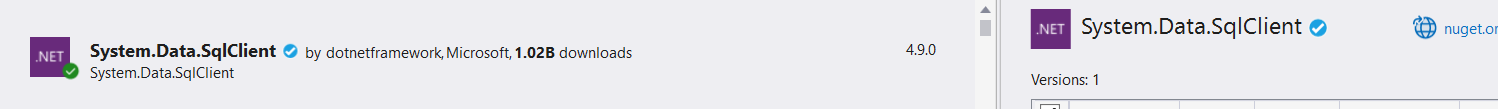
Lấy chuỗi kết nối:

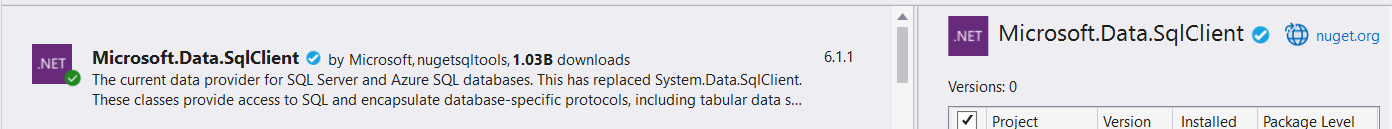


Kiểm tra kết nối :



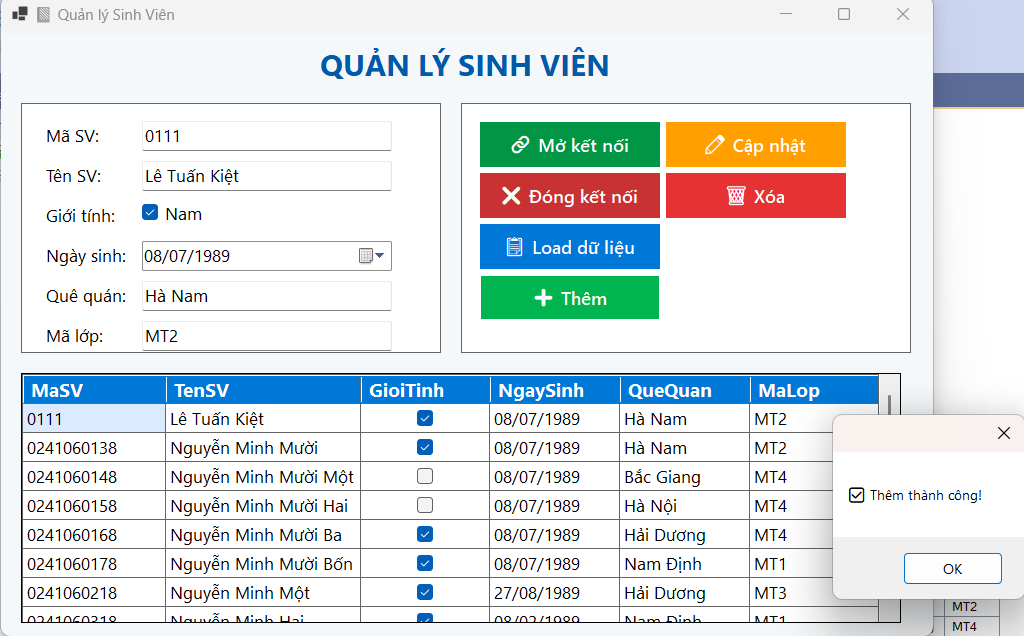
Vào Tools -> Nuget Package Manager tải:



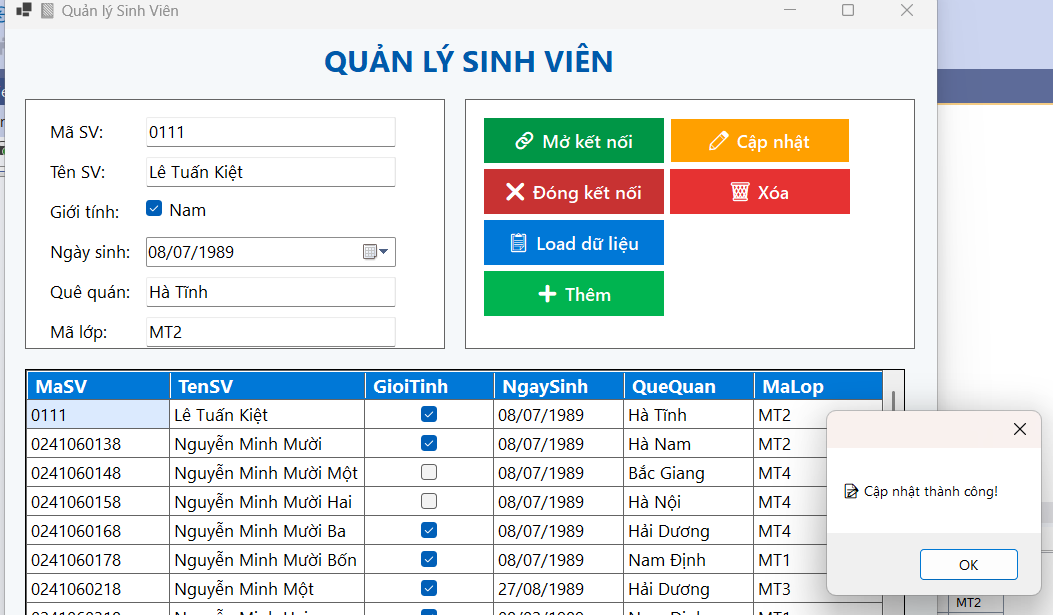




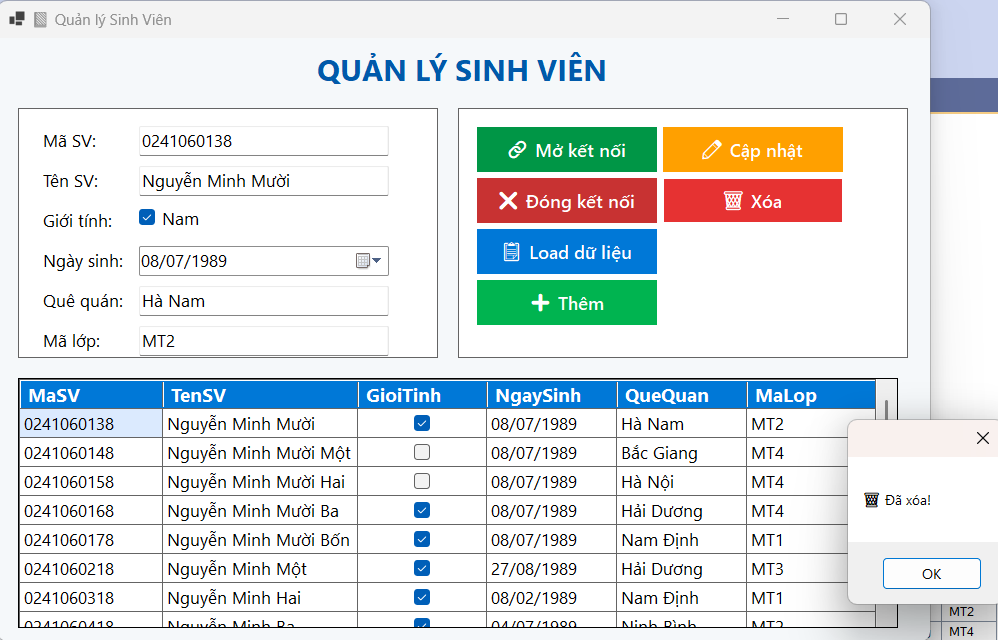
Thêm:



Sửa:



Xóa:



Code trong Form1.cs:

using System;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Windows.Forms;

namespace KetNoiSQL

{

    public partial class Form1 : Form

    {

        string strCon = @"Data Source=TUANKIETLEE\SQLSERVERNEW;Initial Catalog=BTLSQL;Integrated Security=True";

        SqlConnection sqlCon = null;

        SqlDataAdapter adapter = null;

        DataTable dt = null;

        // Controls

        DataGridView dataGridView1;

        TextBox txtMaSV, txtTenSV, txtQueQuan, txtMaLop;

        CheckBox chkGioiTinh;

        DateTimePicker dtNgaySinh;

        Button btnMoKetNoi, btnDongKetNoi, btnLoadData, btnAdd, btnUpdate, btnDelete;

        public Form1()

        {

            InitializeComponent();

            BuildUI();

        }

        private void BuildUI()

        {

            this.Text = "📘 Quản lý Sinh Viên";

            this.Size = new Size(950, 650);

            this.BackColor = Color.FromArgb(245, 248, 250);

            this.Font = new Font("Segoe UI", 10F, FontStyle.Regular);

            // ==== HEADER ====

            Label lblHeader = new Label()

            {

                Text = "QUẢN LÝ SINH VIÊN",

                Font = new Font("Segoe UI", 18, FontStyle.Bold),

                ForeColor = Color.FromArgb(0, 90, 170),

                Dock = DockStyle.Top,

                TextAlign = ContentAlignment.MiddleCenter,

                Height = 60

            };

            this.Controls.Add(lblHeader);

            // ==== PANEL INPUT ====

            Panel pnlInput = new Panel()

            {

                Location = new Point(20, 70),

                Size = new Size(420, 250),

                BackColor = Color.White,

                BorderStyle = BorderStyle.FixedSingle

            };

            this.Controls.Add(pnlInput);

            int y = 20;

            int lblWidth = 100;

            int txtWidth = 250;

            Label lblMaSV = new Label() { Text = "Mã SV:", Location = new Point(20, y), Width = lblWidth };

            txtMaSV = new TextBox() { Location = new Point(120, y - 3), Width = txtWidth };

            y += 40;

            Label lblTenSV = new Label() { Text = "Tên SV:", Location = new Point(20, y), Width = lblWidth };

            txtTenSV = new TextBox() { Location = new Point(120, y - 3), Width = txtWidth };

            y += 40;

            Label lblGioiTinh = new Label() { Text = "Giới tính:", Location = new Point(20, y), Width = lblWidth };

            chkGioiTinh = new CheckBox() { Text = "Nam", Location = new Point(120, y - 3) };

            y += 40;

            Label lblNgaySinh = new Label() { Text = "Ngày sinh:", Location = new Point(20, y), Width = lblWidth };

            dtNgaySinh = new DateTimePicker()

            {

                Location = new Point(120, y - 3),

                Width = txtWidth,

                Format = DateTimePickerFormat.Short

            };

            y += 40;

            Label lblQueQuan = new Label() { Text = "Quê quán:", Location = new Point(20, y), Width = lblWidth };

            txtQueQuan = new TextBox() { Location = new Point(120, y - 3), Width = txtWidth };

            y += 40;

            Label lblMaLop = new Label() { Text = "Mã lớp:", Location = new Point(20, y), Width = lblWidth };

            txtMaLop = new TextBox() { Location = new Point(120, y - 3), Width = txtWidth };

            pnlInput.Controls.AddRange(new Control[]

            {

                lblMaSV, txtMaSV, lblTenSV, txtTenSV,

                lblGioiTinh, chkGioiTinh, lblNgaySinh, dtNgaySinh,

                lblQueQuan, txtQueQuan, lblMaLop, txtMaLop

            });

            // ==== PANEL BUTTONS ====

            FlowLayoutPanel pnlButtons = new FlowLayoutPanel()

            {

                Location = new Point(460, 70),

                Size = new Size(450, 250),

                FlowDirection = FlowDirection.TopDown,

                BackColor = Color.White,

                BorderStyle = BorderStyle.FixedSingle,

                Padding = new Padding(15)

            };

            this.Controls.Add(pnlButtons);

            btnMoKetNoi = CreateButton("🔗 Mở kết nối", Color.FromArgb(0, 150, 70));

            btnDongKetNoi = CreateButton("❌ Đóng kết nối", Color.FromArgb(200, 50, 50));

            btnLoadData = CreateButton("📋 Load dữ liệu", Color.FromArgb(0, 120, 215));

            btnAdd = CreateButton("➕ Thêm", Color.FromArgb(0, 180, 80));

            btnUpdate = CreateButton("✏️ Cập nhật", Color.FromArgb(255, 160, 0));

            btnDelete = CreateButton("🗑️ Xóa", Color.FromArgb(230, 50, 50));

            pnlButtons.Controls.AddRange(new Control[]

            {

                btnMoKetNoi, btnDongKetNoi, btnLoadData, btnAdd, btnUpdate, btnDelete

            });

            // ==== DGV ====

            dataGridView1 = new DataGridView()

            {

                Location = new Point(20, 340),

                Size = new Size(880, 250),

                BackgroundColor = Color.White,

                AutoSizeColumnsMode = DataGridViewAutoSizeColumnsMode.Fill,

                ReadOnly = true

            };

            StyleGrid(dataGridView1);

            this.Controls.Add(dataGridView1);

            // ==== SỰ KIỆN ====

            btnMoKetNoi.Click += btnMoKetNoi\_Click;

            btnDongKetNoi.Click += btnDongKetNoi\_Click;

            btnLoadData.Click += btnLoadData\_Click;

            btnAdd.Click += btnAdd\_Click;

            btnUpdate.Click += btnUpdate\_Click;

            btnDelete.Click += btnDelete\_Click;

            dataGridView1.SelectionChanged += dataGridView1\_SelectionChanged;

            this.FormClosing += Form1\_FormClosing;

        }

        private Button CreateButton(string text, Color color)

        {

            var btn = new Button()

            {

                Text = text,

                Width = 180,

                Height = 45,

                BackColor = color,

                ForeColor = Color.White,

                FlatStyle = FlatStyle.Flat,

                Font = new Font("Segoe UI Semibold", 10.5F)

            };

            btn.FlatAppearance.BorderSize = 0;

            btn.Cursor = Cursors.Hand;

            btn.MouseEnter += (s, e) => { btn.BackColor = ControlPaint.Light(color, 0.3f); };

            btn.MouseLeave += (s, e) => { btn.BackColor = color; };

            return btn;

        }

        private void StyleGrid(DataGridView dgv)

        {

            dgv.EnableHeadersVisualStyles = false;

            dgv.ColumnHeadersDefaultCellStyle.BackColor = Color.FromArgb(0, 120, 215);

            dgv.ColumnHeadersDefaultCellStyle.ForeColor = Color.White;

            dgv.ColumnHeadersDefaultCellStyle.Font = new Font("Segoe UI", 11, FontStyle.Bold);

            dgv.RowHeadersVisible = false;

            dgv.DefaultCellStyle.SelectionBackColor = Color.FromArgb(219, 234, 254);

            dgv.DefaultCellStyle.SelectionForeColor = Color.Black;

        }

        // ================== CODE XỬ LÝ ===================

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

        {

            try

            {

                if (sqlCon == null)

                    sqlCon = new SqlConnection(strCon);

                if (sqlCon.State == ConnectionState.Closed)

                {

                    sqlCon.Open();

                    MessageBox.Show("✅ Kết nối thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

                }

            }

            catch (Exception ex)

            {

                MessageBox.Show("❌ Lỗi: " + ex.Message);

            }

        }

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

        {

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

            {

                sqlCon.Close();

                MessageBox.Show("🔒 Đã đóng kết nối.", "Thông báo");

            }

        }

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

        {

            try

            {

                if (sqlCon == null) sqlCon = new SqlConnection(strCon);

                if (sqlCon.State == ConnectionState.Closed) sqlCon.Open();

                string sql = "SELECT MaSV, TenSV, GioiTinh, NgaySinh, QueQuan, MaLop FROM SinhVien";

                adapter = new SqlDataAdapter(sql, sqlCon);

                dt = new DataTable();

                adapter.Fill(dt);

                dataGridView1.DataSource = dt;

            }

            catch (Exception ex)

            {

                MessageBox.Show("⚠️ Lỗi load dữ liệu: " + ex.Message);

            }

        }

        private void dataGridView1\_SelectionChanged(object sender, EventArgs e)

        {

            if (dataGridView1.CurrentRow == null) return;

            if (dataGridView1.CurrentRow.DataBoundItem is DataRowView drv)

            {

                txtMaSV.Text = drv["MaSV"].ToString();

                txtTenSV.Text = drv["TenSV"].ToString();

                chkGioiTinh.Checked = drv["GioiTinh"].ToString() == "True";

                dtNgaySinh.Value = drv["NgaySinh"] == DBNull.Value ? DateTime.Now : Convert.ToDateTime(drv["NgaySinh"]);

                txtQueQuan.Text = drv["QueQuan"].ToString();

                txtMaLop.Text = drv["MaLop"].ToString();

            }

        }

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

        {

            try

            {

                string sql = "INSERT INTO SinhVien (MaSV, TenSV, GioiTinh, NgaySinh, QueQuan, MaLop) VALUES (@MaSV, @TenSV, @GioiTinh, @NgaySinh, @QueQuan, @MaLop)";

                using (SqlCommand cmd = new SqlCommand(sql, sqlCon))

                {

                    cmd.Parameters.AddWithValue("@MaSV", txtMaSV.Text.Trim());

                    cmd.Parameters.AddWithValue("@TenSV", txtTenSV.Text.Trim());

                    cmd.Parameters.AddWithValue("@GioiTinh", chkGioiTinh.Checked ? 1 : 0);

                    cmd.Parameters.AddWithValue("@NgaySinh", dtNgaySinh.Value);

                    cmd.Parameters.AddWithValue("@QueQuan", txtQueQuan.Text.Trim());

                    cmd.Parameters.AddWithValue("@MaLop", txtMaLop.Text.Trim());

                    cmd.ExecuteNonQuery();

                }

                btnLoadData\_Click(null, null);

                MessageBox.Show("✅ Thêm thành công!");

            }

            catch (Exception ex)

            {

                MessageBox.Show("❌ Lỗi thêm: " + ex.Message);

            }

        }

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

        {

            try

            {

                string sql = "UPDATE SinhVien SET TenSV=@TenSV, GioiTinh=@GioiTinh, NgaySinh=@NgaySinh, QueQuan=@QueQuan, MaLop=@MaLop WHERE MaSV=@MaSV";

                using (SqlCommand cmd = new SqlCommand(sql, sqlCon))

                {

                    cmd.Parameters.AddWithValue("@MaSV", txtMaSV.Text.Trim());

                    cmd.Parameters.AddWithValue("@TenSV", txtTenSV.Text.Trim());

                    cmd.Parameters.AddWithValue("@GioiTinh", chkGioiTinh.Checked ? 1 : 0);

                    cmd.Parameters.AddWithValue("@NgaySinh", dtNgaySinh.Value);

                    cmd.Parameters.AddWithValue("@QueQuan", txtQueQuan.Text.Trim());

                    cmd.Parameters.AddWithValue("@MaLop", txtMaLop.Text.Trim());

                    cmd.ExecuteNonQuery();

                }

                btnLoadData\_Click(null, null);

                MessageBox.Show("📝 Cập nhật thành công!");

            }

            catch (Exception ex)

            {

                MessageBox.Show("⚠️ Lỗi cập nhật: " + ex.Message);

            }

        }

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

        {

            try

            {

                string sql = "DELETE FROM SinhVien WHERE MaSV=@MaSV";

                using (SqlCommand cmd = new SqlCommand(sql, sqlCon))

                {

                    cmd.Parameters.AddWithValue("@MaSV", txtMaSV.Text.Trim());

                    cmd.ExecuteNonQuery();

                }

                btnLoadData\_Click(null, null);

                MessageBox.Show("🗑️ Đã xóa!");

            }

            catch (Exception ex)

            {

                MessageBox.Show("❌ Lỗi xóa: " + ex.Message);

            }

        }

        private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

        {

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

                sqlCon.Close();

        }

    }

}