**THỰC HÀNH BUỔI 6**

Họ và tên: Đinh Thị Thảo An

Lớp: 11\_ĐH\_CNPM1

Mssv: 1150080083

Mục lục

[**Lab3** 3](#_Toc210834209)

[**Thực hành 1:** 3](#_Toc210834210)

[Thực hành 2: 12](#_Toc210834211)

[Lab 4 20](#_Toc210834212)

[Thực hành 1: Tạo kết nối giữa Winform và CSDL. 20](#_Toc210834213)

[Thực hành 2: Truy vấn dữ liệu 25](#_Toc210834214)

[1. Truy vấn lấy 1 giá trị 25](#_Toc210834215)

[**2. Truy vấn lấy 1 dòng dữ liệu** 28](#_Toc210834216)

[**3. Truy vấn lấy nhiều dòng dữ liệu** 32](#_Toc210834217)

[4. Sử dụng Parameter để truy vấn dữ liệu trong C# 36](#_Toc210834218)

[**Áp dụng: Hiển thị danh sách sinh viên theo mã lớp.** 40](#_Toc210834219)

# 

# **Lab3**

## **Thực hành 1:**

**Code:**

using System;

using System.IO;

using System.Text;

using System.Windows.Forms;

using System.Drawing;

namespace DinhThiThaoAn\_1150080083\_Lab3

{

public partial class ThucHanh1 : Form

{

public ThucHanh1()

{

InitializeComponent();

// Nạp ảnh logo (thử 2 vị trí phổ biến)

string exeDir = AppDomain.CurrentDomain.BaseDirectory;

string p1 = Path.Combine(exeDir, "bago.jpg");

string p2 = Path.Combine(exeDir, "Images", "bago.jpg");

string imgPath = File.Exists(p1) ? p1 : (File.Exists(p2) ? p2 : null);

if (imgPath != null)

{

// Cách không khóa file (để tránh lỗi copy khi rebuild)

byte[] bytes = File.ReadAllBytes(imgPath);

using (var ms = new MemoryStream(bytes))

using (var tmp = Image.FromStream(ms))

{

picLogo.Image = new Bitmap(tmp);

}

picLogo.SizeMode = PictureBoxSizeMode.Zoom;

}

else

{

picLogo.BackColor = Color.LightYellow;

}

// Nạp danh sách bàn

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

if (cboBan.Items.Count > 0) cboBan.SelectedIndex = 0;

// Gắn handler cho các nút món

btnComChienTrung.Click += OnMenuButtonClick;

btnBanhMiOpLa.Click += OnMenuButtonClick;

btnCoca.Click += OnMenuButtonClick;

btnLipton.Click += OnMenuButtonClick;

btnOcRangMuoi.Click += OnMenuButtonClick;

btnKhoaiTayChien.Click += OnMenuButtonClick;

btn7Up.Click += OnMenuButtonClick;

btnCam.Click += OnMenuButtonClick;

btnMyXaoHaiSan.Click += OnMenuButtonClick;

btnCaVienChien.Click += OnMenuButtonClick;

btnPepsi.Click += OnMenuButtonClick;

btnCafe.Click += OnMenuButtonClick;

btnBurgerBoNuong.Click += OnMenuButtonClick;

btnDuiGaRan.Click += OnMenuButtonClick;

btnBunBoHue.Click += OnMenuButtonClick;

// Nút chức năng

btnXoa.Click += BtnXoa\_Click;

btnOrder.Click += BtnOrder\_Click;

// Cấu hình ListView

lvOrder.View = View.Details;

lvOrder.FullRowSelect = true;

lvOrder.GridLines = true;

if (lvOrder.Columns.Count == 0)

{

lvOrder.Columns.Add("Món", 230);

lvOrder.Columns.Add("Số lượng", 90, HorizontalAlignment.Right);

}

}

private void OnMenuButtonClick(object sender, EventArgs e)

{

var btn = sender as Button;

if (btn == null) return;

AddOrIncreaseItem(btn.Text.Trim(), 1);

}

private void AddOrIncreaseItem(string tenMon, int soLuongTang)

{

foreach (ListViewItem it in lvOrder.Items)

{

if (string.Equals(it.SubItems[0].Text, tenMon, StringComparison.CurrentCultureIgnoreCase))

{

int sl = int.Parse(it.SubItems[1].Text);

it.SubItems[1].Text = (sl + soLuongTang).ToString();

return;

}

}

var item = new ListViewItem(tenMon);

item.SubItems.Add(soLuongTang.ToString());

lvOrder.Items.Add(item);

}

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

{

if (lvOrder.SelectedItems.Count == 0)

{

MessageBox.Show("Hãy chọn món trong danh sách để xóa.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

foreach (ListViewItem it in lvOrder.SelectedItems)

lvOrder.Items.Remove(it);

}

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

{

if (lvOrder.Items.Count == 0)

{

MessageBox.Show("Danh sách món đang trống.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

if (cboBan.SelectedItem == null)

{

MessageBox.Show("Vui lòng chọn bàn.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

string tenBan = cboBan.SelectedItem.ToString();

string dir = Path.Combine(Environment.GetFolderPath(Environment.SpecialFolder.Desktop), "Orders");

Directory.CreateDirectory(dir);

string fileName = $"Order\_{tenBan.Replace(' ', '\_')}\_{DateTime.Now:yyyyMMdd\_HHmmss}.txt";

string fullPath = Path.Combine(dir, fileName);

var sb = new StringBuilder();

sb.AppendLine("QUÁN ĂN NHANH HÙNG THỊNH");

sb.AppendLine($"Bàn: {tenBan}");

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

sb.AppendLine(new string('-', 40));

sb.AppendLine($"{"Món",-28}{"SL",3}");

sb.AppendLine(new string('-', 40));

foreach (ListViewItem it in lvOrder.Items)

{

string mon = it.SubItems[0].Text;

string sl = it.SubItems[1].Text;

string monTrim = mon.Length > 28 ? mon.Substring(0, 28) : mon;

sb.AppendLine($"{monTrim,-28}{sl,3}");

}

sb.AppendLine(new string('-', 40));

sb.AppendLine("Ghi chú: File tạo tự động, thay cho gửi xuống bếp.");

File.WriteAllText(fullPath, sb.ToString(), new UTF8Encoding(true));

MessageBox.Show($"Đã lưu đơn hàng:\n{fullPath}", "Thành công",

MessageBoxButtons.OK, MessageBoxIcon.Information);

lvOrder.Items.Clear();

}

}

}

**Kết quả**

Order

A screenshot of a computer

AI-generated content may be incorrect.

Xóa cơm chiên trứng

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Chọn bàn

A screenshot of a computer

AI-generated content may be incorrect.

## Thực hành 2:

**Code:**

using System;

using System.Windows.Forms;

namespace DinhThiThaoAn\_1150080083\_Lab3

{

public partial class ThucHanh2 : Form

{

public ThucHanh2()

{

InitializeComponent();

// Sự kiện

btnThem.Click += BtnThem\_Click;

btnSua.Click += BtnSua\_Click;

btnXoa.Click += BtnXoa\_Click;

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

lvSinhVien.SelectedIndexChanged += LvSinhVien\_SelectedIndexChanged;

}

// Thêm: kiểm tra Họ tên không rỗng

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

{

string hoTen = txtHoTen.Text.Trim();

if (string.IsNullOrEmpty(hoTen))

{

MessageBox.Show("Họ tên không được để trống.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtHoTen.Focus();

return;

}

string lop = txtLop.Text.Trim();

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

string diaChi = txtDiaChi.Text.Trim();

var item = new ListViewItem(hoTen);

item.SubItems.Add(ngaySinh);

item.SubItems.Add(lop);

item.SubItems.Add(diaChi);

lvSinhVien.Items.Add(item);

ClearInputs();

}

// Xóa: yêu cầu chọn dòng

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

{

if (lvSinhVien.SelectedItems.Count == 0)

{

MessageBox.Show("Hãy chọn một dòng để xóa.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

foreach (ListViewItem it in lvSinhVien.SelectedItems)

lvSinhVien.Items.Remove(it);

ClearInputs();

}

// Sửa: cập nhật dòng đang chọn

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

{

if (lvSinhVien.SelectedItems.Count == 0)

{

MessageBox.Show("Hãy chọn một dòng để sửa.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

var sel = lvSinhVien.SelectedItems[0];

string hoTen = txtHoTen.Text.Trim();

if (string.IsNullOrEmpty(hoTen))

{

MessageBox.Show("Họ tên không được để trống.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtHoTen.Focus();

return;

}

sel.SubItems[0].Text = hoTen;

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

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

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

}

// Chọn dòng -> đổ dữ liệu lên vùng nhập

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

{

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

var it = lvSinhVien.SelectedItems[0];

txtHoTen.Text = it.SubItems[0].Text;

DateTime d;

if (DateTime.TryParse(it.SubItems[1].Text, out d))

dtpNgaySinh.Value = d;

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

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

}

private void ClearInputs()

{

txtHoTen.Clear();

txtLop.Clear();

txtDiaChi.Clear();

dtpNgaySinh.Value = DateTime.Today;

txtHoTen.Focus();

}

}

}

**Kết quả:**

Thêm thông tin sinh viên

A screenshot of a calendar

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

* Sửa Ha từ 10/06/2025 thành 03/10/2025

A screenshot of a computer

AI-generated content may be incorrect.

# Lab 4

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

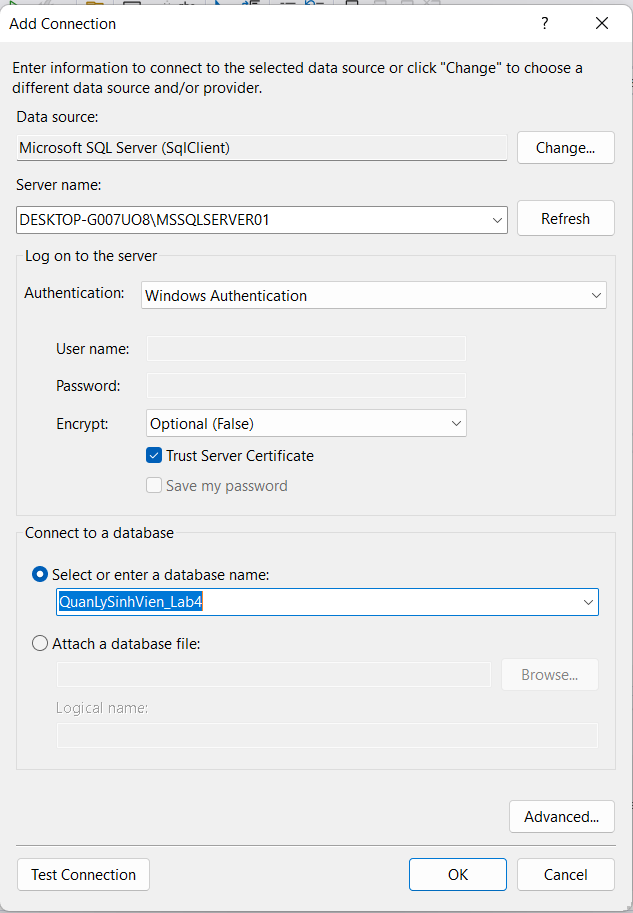
A screenshot of a computer

AI-generated content may be incorrect.

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

A screenshot of a computer

AI-generated content may be incorrect.



Paste Server name vào (1) → chọn databse đã có trên máy local trong (2)

Nhấn (3) để lấy chuỗi kết nối:

A screenshot of a computer

AI-generated content may be incorrect.

Nhấn (4) để kiểm tra kết nối:

A screenshot of a computer error

AI-generated content may be incorrect.

Nhấn (5) để hoàn tất kết nối!

A screenshot of a computer

AI-generated content may be incorrect.

Kiểm tra kết nối  
code:

string strCon = @"Data Source=DESKTOP-G007UO8\MSSQLSERVER01;Initial Catalog=QuanLySinhVien\_Lab4;Integrated Security=True";

SqlConnection sqlCon = null;

public Form1()

{

InitializeComponent();

}

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

{

try

{

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

if (sqlCon.State == System.Data.ConnectionState.Closed)

{

sqlCon.Open();

MessageBox.Show("Kết nối thành công!");

}

}

catch (Exception ex)

{

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

}

}

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

{

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

{

sqlCon.Close();

MessageBox.Show("Đã đóng kết nối!");

}

else

{

MessageBox.Show("Chưa tạo kết nối!");

}

}

## Thực hành 2: Truy vấn dữ liệu

### 1. Truy vấn lấy 1 giá trị

**Code:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Lab4\_WinformKetNoiCSDL

{

public partial class Form2 : Form

{

string strCon = @"Data Source=DESKTOP-G007UO8\MSSQLSERVER01;Initial Catalog=QuanLySinhVien\_Lab4;Integrated Security=True";

SqlConnection sqlCon = null;

public Form2()

{

InitializeComponent();

}

private void Form2\_Load(object sender, EventArgs e)

{

}

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

{

{

try

{

if (sqlCon == null)

sqlCon = new SqlConnection(strCon);

if (sqlCon.State == ConnectionState.Closed)

sqlCon.Open();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "SELECT COUNT(\*) FROM SINHVIEN";

sqlCmd.Connection = sqlCon;

int soLuongSV = (int)sqlCmd.ExecuteScalar();

MessageBox.Show("Số lượng sinh viên là: " + soLuongSV,

"Kết quả", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception ex)

{

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

}

}

}

}

}

**Kết quả:**

**A screenshot of a computer

AI-generated content may be incorrect.**

### **2. Truy vấn lấy 1 dòng dữ liệu**

**Code:**

using System;

using System.Data;

using System.Data.SqlClient;

using System.Windows.Forms;

namespace Lab4\_WinformKetNoiCSDL

{

public partial class Form1 : Form

{

string strCon = @"Data Source=DESKTOP-G007UO8\MSSQLSERVER01;Initial Catalog=QuanLySinhVien\_Lab4;Integrated Security=True";

SqlConnection sqlCon = null;

public Form1()

{

InitializeComponent();

}

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

{

try

{

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

if (sqlCon.State == System.Data.ConnectionState.Closed)

{

sqlCon.Open();

MessageBox.Show("Kết nối thành công!");

}

}

catch (Exception ex)

{

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

}

}

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

{

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

{

sqlCon.Close();

MessageBox.Show("Đã đóng kết nối!");

}

else

{

MessageBox.Show("Chưa tạo kết nối!");

}

}

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

{

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

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

var maSV = txtNhapMaSV.Text.Trim();

if (string.IsNullOrWhiteSpace(maSV))

{

MessageBox.Show("Vui lòng nhập mã sinh viên!");

return;

}

using (var cmd = new SqlCommand("SELECT \* FROM SINHVIEN WHERE MaSV=@maSV", sqlCon))

{

cmd.CommandType = CommandType.Text;

cmd.Parameters.AddWithValue("@maSV", maSV);

using (var r = cmd.ExecuteReader())

{

if (r.Read())

{

txtTenSV.Text = r.GetString(1);

txtGioiTinh.Text = r.GetString(2);

txtNgaySinh.Text = r.GetDateTime(3).ToString("dd/MM/yyyy");

txtQueQuan.Text = r.GetString(4);

txtMaLop.Text = r.GetString(5);

}

else

{

MessageBox.Show("Không tìm thấy sinh viên!");

txtTenSV.Clear(); txtGioiTinh.Clear(); txtNgaySinh.Clear();

txtQueQuan.Clear(); txtMaLop.Clear();

}

}

}

}

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

{

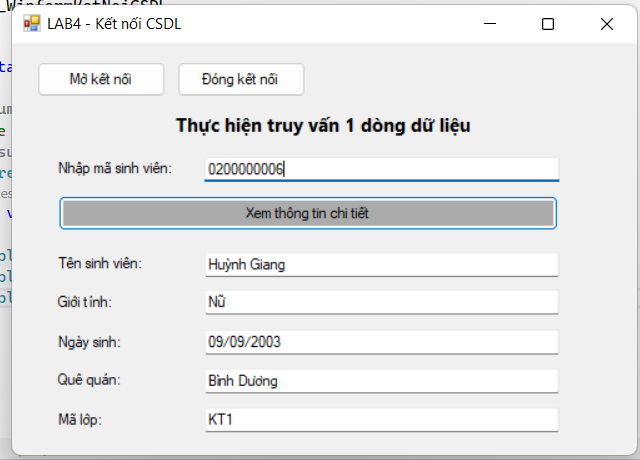
}

}

}

**Kết quả**

Nhập mã số sinh viên “0200000006” và bấm xem thông tin chi tiết, sẽ ra kết quả sau

****

### **3. Truy vấn lấy nhiều dòng dữ liệu**

**Code**

using System;

using System.Data;

using System.Data.SqlClient;

using System.Windows.Forms;

namespace Lab4\_WinformKetNoiCSDL

{

public partial class Form3 : Form

{

string strCon = @"Data Source=DESKTOP-G007UO8\MSSQLSERVER01;Initial Catalog=QuanLySinhVien\_Lab4;Integrated Security=True";

SqlConnection sqlCon = null;

public Form3()

{

InitializeComponent();

}

private void EnsureOpen()

{

if (sqlCon == null)

sqlCon = new SqlConnection(strCon);

if (sqlCon.State == ConnectionState.Closed)

sqlCon.Open();

}

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

{

try

{

EnsureOpen();

lsvList.Items.Clear();

using (SqlCommand cmd = new SqlCommand("SELECT \* FROM SINHVIEN", sqlCon))

{

cmd.CommandType = CommandType.Text;

using (SqlDataReader reader = cmd.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);

lsvList.Items.Add(lvi);

}

}

}

}

catch (Exception ex)

{

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

}

}

protected override void OnFormClosing(FormClosingEventArgs e)

{

try { if (sqlCon != null && sqlCon.State == ConnectionState.Open) sqlCon.Close(); }

catch { }

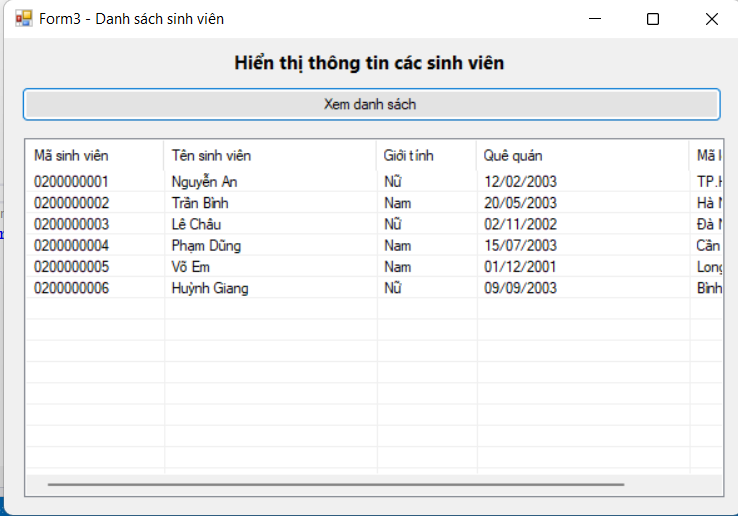
base.OnFormClosing(e);

}

}

}

**Kết quả**



### 4. Sử dụng Parameter để truy vấn dữ liệu trong C#

**Code**

using System;

using System.Data;

using System.Data.SqlClient;

using System.Windows.Forms;

namespace Lab4\_WinformKetNoiCSDL

{

public partial class Form4 : Form

{

private string strCon = @"Data Source=DESKTOP-G007UO8\MSSQLSERVER01;Initial Catalog=QuanLySinhVien\_Lab4;Integrated Security=True";

private SqlConnection sqlCon = null;

public Form4()

{

InitializeComponent();

}

private void EnsureOpen()

{

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

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

}

private string MapTenKhoaToMa(string tenKhoa)

{

tenKhoa = tenKhoa.Trim();

if (tenKhoa.Equals("Công nghệ thông tin", StringComparison.OrdinalIgnoreCase)) return "CNTT";

if (tenKhoa.Equals("Cơ khí", StringComparison.OrdinalIgnoreCase)) return "CK";

if (tenKhoa.Equals("Điện tử", StringComparison.OrdinalIgnoreCase)) return "DT";

if (tenKhoa.Equals("Kinh tế", StringComparison.OrdinalIgnoreCase)) return "KT";

return string.Empty;

}

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

{

try

{

EnsureOpen();

string tenKhoa = txtNhapTenKhoa.Text.Trim();

string maKhoa = MapTenKhoaToMa(tenKhoa);

if (string.IsNullOrEmpty(maKhoa))

{

MessageBox.Show("Tên khoa không hợp lệ. Nhập: Công nghệ thông tin / Cơ khí / Điện tử / Kinh tế.");

return;

}

lsvDanhSach.Items.Clear();

using (var cmd = new SqlCommand("SELECT \* FROM LOP WHERE MaKhoa=@maKhoa", sqlCon))

{

cmd.CommandType = CommandType.Text;

cmd.Parameters.Add(new SqlParameter("@maKhoa", SqlDbType.Char) { Value = maKhoa });

using (var r = cmd.ExecuteReader())

{

while (r.Read())

{

string maLop = r.GetString(0);

string tenLop = r.GetString(1);

var lvi = new ListViewItem(tenLop);

lvi.SubItems.Add(maLop);

lsvDanhSach.Items.Add(lvi);

}

}

}

}

catch (Exception ex)

{

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

}

}

protected override void OnFormClosing(FormClosingEventArgs e)

{

try { if (sqlCon != null && sqlCon.State == ConnectionState.Open) sqlCon.Close(); } catch { }

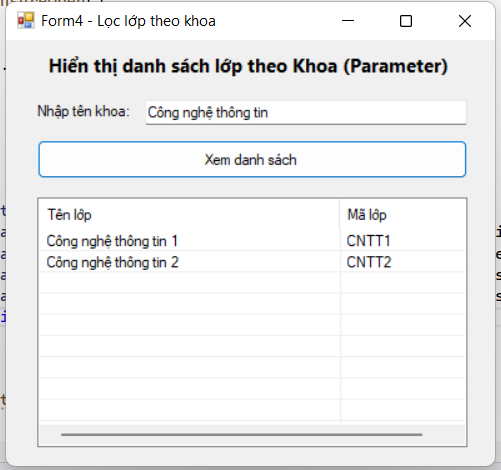
base.OnFormClosing(e);

}

}

}

**Kết quả**

****

### **Áp dụng: Hiển thị danh sách sinh viên theo mã lớp.**

**Code**

using System;

using System.Data;

using System.Data.SqlClient;

using System.Windows.Forms;

namespace Lab4\_WinformKetNoiCSDL

{

public partial class Form5 : Form

{

private string strCon = @"Data Source=DESKTOP-G007UO8\MSSQLSERVER01;Initial Catalog=QuanLySinhVien\_Lab4;Integrated Security=True";

private SqlConnection sqlCon = null;

public Form5()

{

InitializeComponent();

}

private void EnsureOpen()

{

if (sqlCon == null)

sqlCon = new SqlConnection(strCon);

if (sqlCon.State == ConnectionState.Closed)

sqlCon.Open();

}

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

{

try

{

EnsureOpen();

string maLop = txtMaLop.Text.Trim();

if (string.IsNullOrEmpty(maLop))

{

MessageBox.Show("Vui lòng nhập mã lớp (ví dụ: CNTT1, CK1, KT1...)");

return;

}

lsvSinhVien.Items.Clear();

SqlCommand cmd = new SqlCommand();

cmd.CommandType = CommandType.Text;

cmd.CommandText = "SELECT \* FROM SinhVien WHERE MaLop = @MaLop";

cmd.Connection = sqlCon;

cmd.Parameters.Add(new SqlParameter("@MaLop", SqlDbType.Char) { Value = maLop });

SqlDataReader reader = cmd.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 maL = 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(maL);

lsvSinhVien.Items.Add(lvi);

}

reader.Close();

}

catch (Exception ex)

{

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

}

}

protected override void OnFormClosing(FormClosingEventArgs e)

{

try { if (sqlCon != null && sqlCon.State == ConnectionState.Open) sqlCon.Close(); } catch { }

base.OnFormClosing(e);

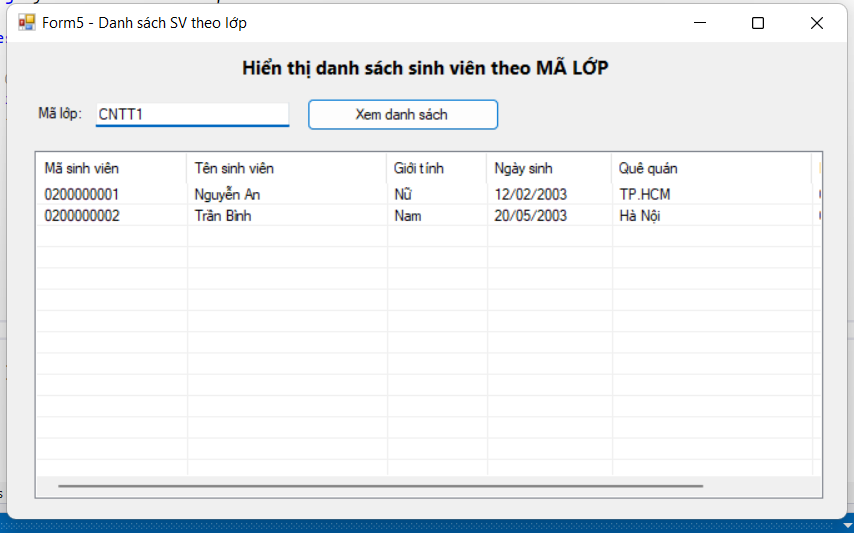
}

}

}

**Kết quả**

Xem danh sách sinh viên theo mã lớp “CNTT1”

****