**LAB 3**

**THỰC HÀNH 1**

|  |  |
| --- | --- |
|  | **Khi order thành công** |

**Code:**

**Form1.cs**

using System;

using System.Collections.Generic;

using System.IO;

using System.Windows.Forms;

namespace QuanAnNhanh

{

public partial class Form1 : Form

{

private readonly Dictionary<string, int> orders =

new Dictionary<string, int>(StringComparer.CurrentCulture);

public Form1()

{

InitializeComponent(); // KHÔNG tạo nút ở đây để tránh lỗi Designer

}

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

{

if (sender is Button b)

{

string tenMon = b.Text;

if (orders.ContainsKey(tenMon)) orders[tenMon]++;

else orders[tenMon] = 1;

CapNhatGrid();

}

}

private void CapNhatGrid()

{

dgvOrders.Rows.Clear();

foreach (var kv in orders)

dgvOrders.Rows.Add(kv.Key, kv.Value);

}

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

{

orders.Clear();

dgvOrders.Rows.Clear();

}

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

{

if (cbBan.SelectedItem == null)

{

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

MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

if (orders.Count == 0)

{

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

MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

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

using (var sw = new StreamWriter(fileName, true))

{

sw.WriteLine("========== ĐƠN GỌI MÓN ==========");

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

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

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

foreach (var kv in orders)

sw.WriteLine($"{kv.Key,-25} x{kv.Value}");

sw.WriteLine();

}

MessageBox.Show($"Đã lưu đơn hàng vào file {fileName}",

"Thành công", MessageBoxButtons.OK, MessageBoxIcon.Information);

orders.Clear();

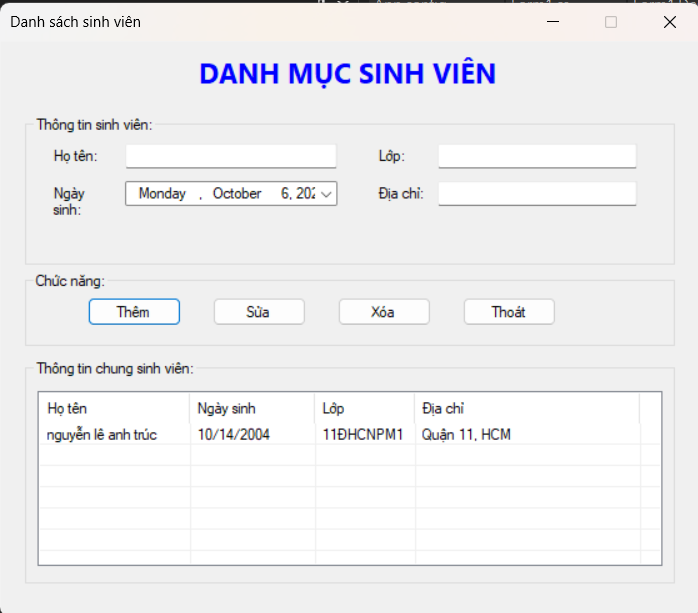
dgvOrders.Rows.Clear();

}

}

}

**THỰC HÀNH 2**

****

**Code:**

**Form1.cs**

using System;

using System.Windows.Forms;

namespace ThucHanh2

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

// Nút Thêm

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

{

if (string.IsNullOrWhiteSpace(txtHoTen.Text))

{

MessageBox.Show("Họ tên sinh viên không được để trống!", "Lỗi",

MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtHoTen.Focus();

return;

}

ListViewItem item = new ListViewItem(txtHoTen.Text);

item.SubItems.Add(dtNgaySinh.Value.ToShortDateString());

item.SubItems.Add(txtLop.Text);

item.SubItems.Add(txtDiaChi.Text);

lvSinhVien.Items.Add(item);

ClearInput();

}

// Nút Xóa

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

{

if (lvSinhVien.SelectedItems.Count == 0)

{

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

MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

if (MessageBox.Show("Bạn có chắc chắn muốn xóa dòng này?", "Xác nhận",

MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

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

ClearInput();

}

}

// Nút Sửa

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

{

if (lvSinhVien.SelectedItems.Count == 0)

{

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

MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

var item = lvSinhVien.SelectedItems[0];

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

item.SubItems[1].Text = dtNgaySinh.Value.ToShortDateString();

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

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

MessageBox.Show("Cập nhật thành công!", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

ClearInput();

}

// Nút Thoát

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

{

if (MessageBox.Show("Bạn có chắc muốn thoát?", "Thoát",

MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

Application.Exit();

}

}

// Khi chọn dòng trong ListView

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

{

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

var item = lvSinhVien.SelectedItems[0];

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

dtNgaySinh.Text = item.SubItems[1].Text;

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

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

}

// Xóa dữ liệu nhập

private void ClearInput()

{

txtHoTen.Clear();

txtLop.Clear();

txtDiaChi.Clear();

dtNgaySinh.Value = DateTime.Now;

}

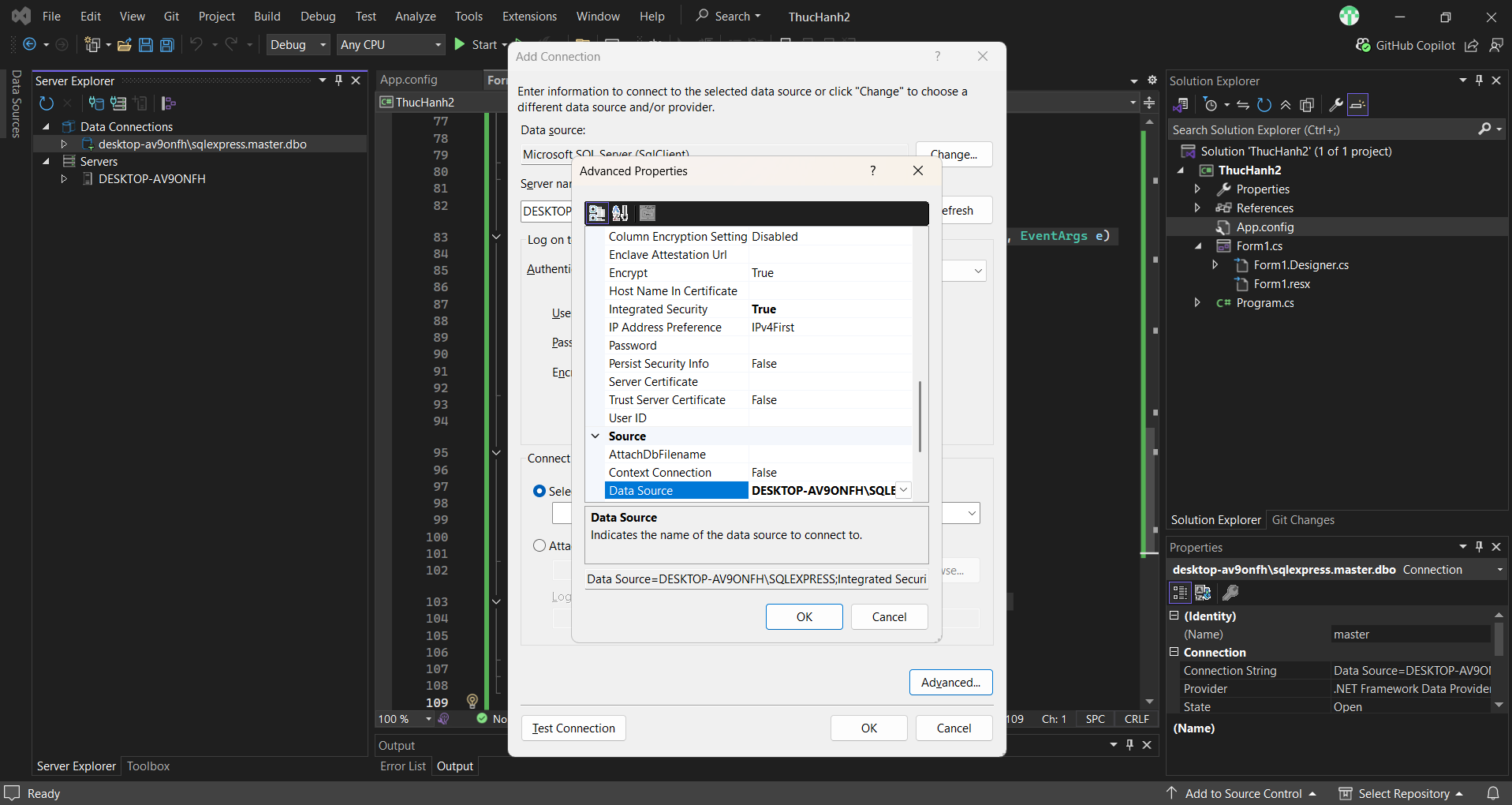
}

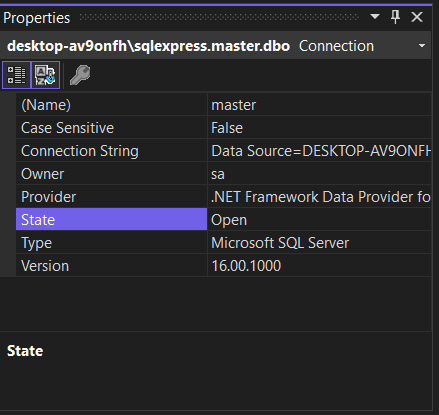
}

**LAB 4**

**THỰC HÀNH 1**

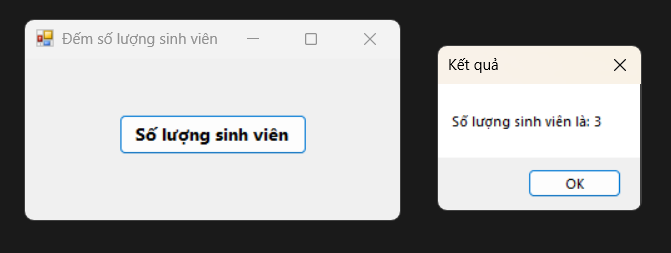
**Kết nối dữ liệu**

****

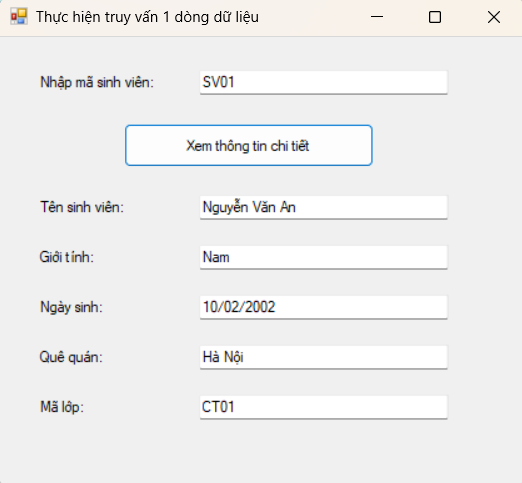
****

**THỰC HÀNH 2**

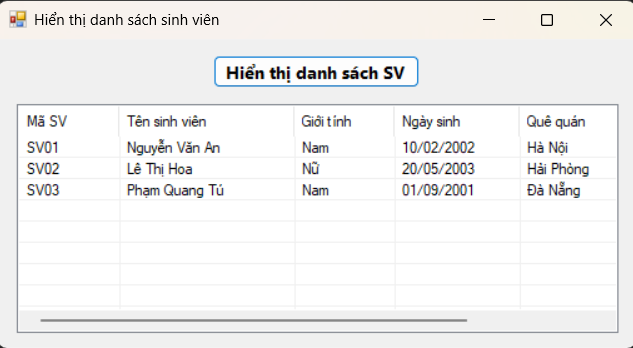
**1. Truy vấn lấy 1 giá trị**

****

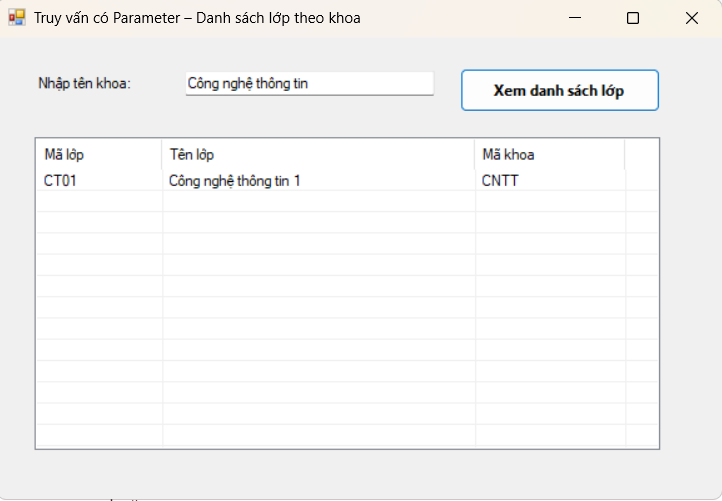
**2. Truy vấn lấy 1 dòng dữ liệu**

****

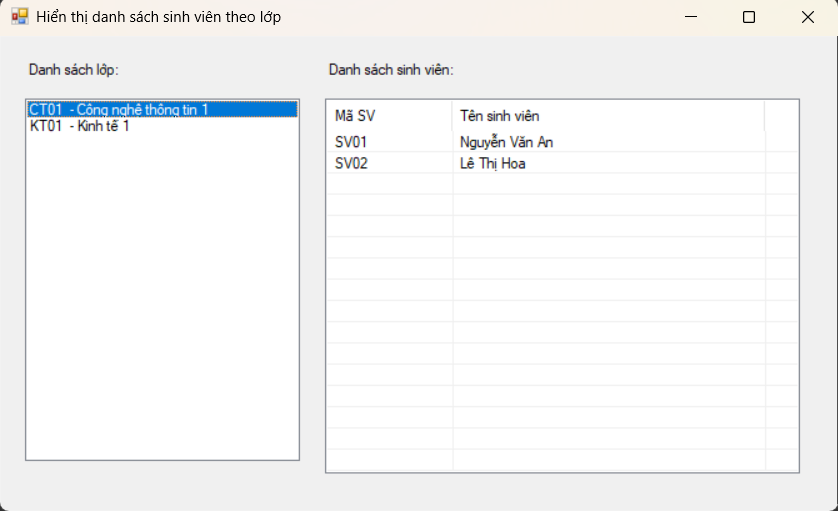
**3. Truy vấn lấy nhiều dòng dữ liệu**

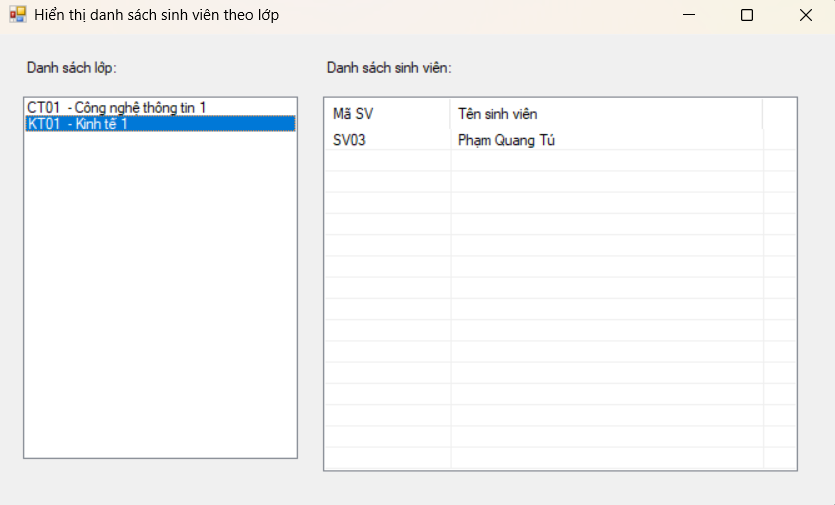
****

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

****

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

****

****