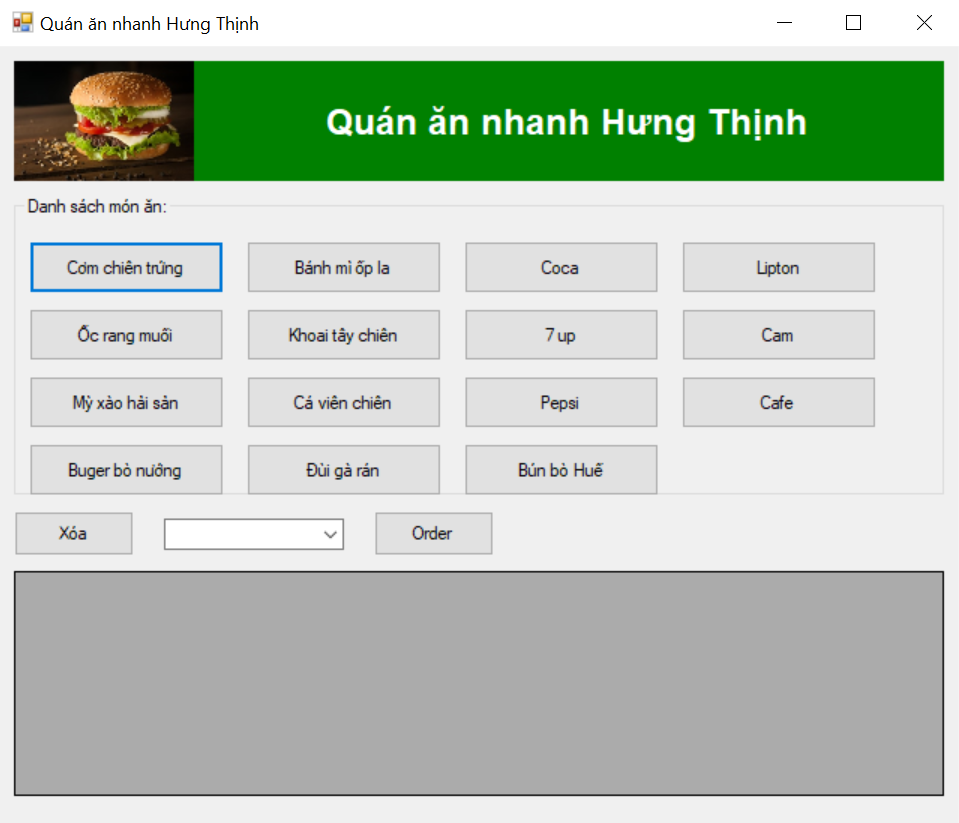
**Lab 3**

**Thực hành 1**



Code xử lí logic

using System;

using System.Collections.Generic;

using System.Data;

using System.IO;

using System.Linq;

using System.Net;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace NguyenHoangKy\_1150080023\_CNPM1\_Bt5

{

public partial class Form4 : Form

{

// Danh sách lưu món ăn và số lượng

private Dictionary<string, int> danhSachMon = new Dictionary<string, int>();

public Form4()

{

InitializeComponent();

// Gán sự kiện Load để tải ảnh dynamic

this.Load += new EventHandler(this.Form4\_Load);

TaoSuKienChoCacButtonMonAn();

// Gán sự kiện cho btnXoa và btnOrder

btnXoa.Click += btnXoa\_Click;

btnOrder.Click += btnOrder\_Click;

}

// Sự kiện Load form để tải ảnh

private async void Form4\_Load(object sender, EventArgs e)

{

await TaiAnhTuURL();

}

// Tải ảnh từ URL dynamic

private async Task TaiAnhTuURL()

{

string urlAnh = "https://images.unsplash.com/photo-1571091718767-18b5b1457add?w=300&fit=crop"; // URL direct cheeseburger free

try

{

using (WebClient client = new WebClient())

{

byte[] imageBytes = await client.DownloadDataTaskAsync(urlAnh);

using (var ms = new System.IO.MemoryStream(imageBytes))

{

pictureBox1.Image = System.Drawing.Image.FromStream(ms);

}

}

}

catch (Exception ex)

{

MessageBox.Show("Lỗi tải ảnh: " + ex.Message + "\nKiểm tra kết nối mạng!");

// Fallback: Có thể gán ảnh default từ Resources nếu có

}

}

private void TaoSuKienChoCacButtonMonAn()

{

// Duyệt qua tất cả Button trong groupBox1

foreach (Control c in groupBox1.Controls)

{

if (c is Button btn)

{

btn.Click += BtnMonAn\_Click;

}

}

}

// Khi nhấn vào món ăn

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

{

Button btn = sender as Button;

string tenMon = btn.Text;

// Nếu món ăn đã có thì tăng số lượng, ngược lại thêm mới

if (danhSachMon.ContainsKey(tenMon))

danhSachMon[tenMon]++;

else

danhSachMon[tenMon] = 1;

CapNhatDataGridView();

}

// Cập nhật hiển thị DataGridView

private void CapNhatDataGridView()

{

DataTable dt = new DataTable();

dt.Columns.Add("Tên món");

dt.Columns.Add("Số lượng");

foreach (var mon in danhSachMon)

{

dt.Rows.Add(mon.Key, mon.Value);

}

dataGridView1.DataSource = dt;

}

// Xóa danh sách

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

{

danhSachMon.Clear();

CapNhatDataGridView();

}

// Ghi tệp Order

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

{

if (cboChonBan.SelectedItem == null)

{

MessageBox.Show("Vui lòng chọn bàn trước khi order!");

return;

}

string tenBan = cboChonBan.SelectedItem.ToString();

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

using (StreamWriter sw = new StreamWriter(fileName))

{

sw.WriteLine($"Bàn: {tenBan}");

sw.WriteLine($"Thời gian: {DateTime.Now}");

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

foreach (var mon in danhSachMon)

{

sw.WriteLine($"{mon.Key}: {mon.Value}");

}

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

sw.WriteLine("Trạng thái: Đã gửi xuống bếp");

}

MessageBox.Show($"Đã ghi order của {tenBan} vào file:\n{fileName}");

// Clear danh sách sau khi order

danhSachMon.Clear();

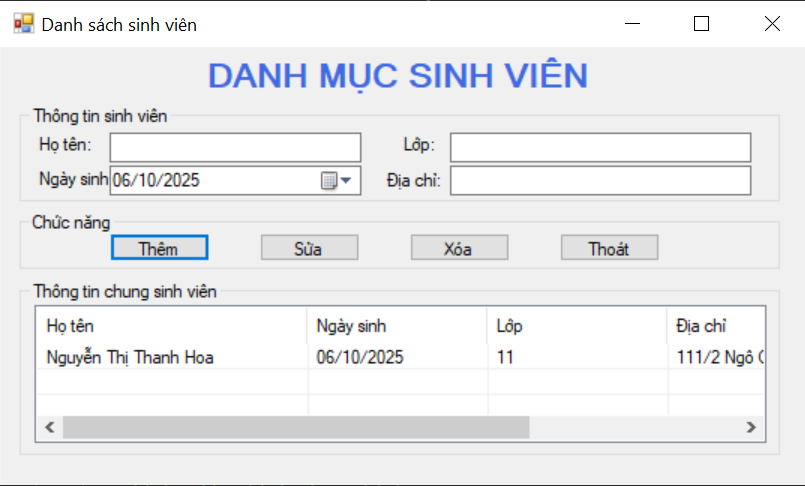
CapNhatDataGridView();

}

}

}

**Thực hành 2**

****

**Code xử lí logic**

using System;

using System.Windows.Forms;

namespace NguyenHoangKy\_1150080023\_CNPM1\_Bt5

{

public partial class Form5 : Form

{

public Form5()

{

InitializeComponent();

}

private void Form5\_Load(object sender, EventArgs e)

{

// Có thể khởi tạo dữ liệu mẫu nếu muốn

}

// 🟢 Nút THÊM

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

{

string hoTen = txtHoTen.Text.Trim();

string lop = txtLop.Text.Trim();

string diaChi = txtDiaChi.Text.Trim();

string ngaySinh = dtpNgaySinh.Value.ToShortDateString();

if (string.IsNullOrEmpty(hoTen))

{

MessageBox.Show("Họ tên sinh viên không được để trống!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

txtHoTen.Focus();

return;

}

ListViewItem item = new ListViewItem(hoTen);

item.SubItems.Add(ngaySinh);

item.SubItems.Add(lop);

item.SubItems.Add(diaChi);

lvSinhVien.Items.Add(item);

ClearTextBoxes();

}

// 🟠 Nút SỬA

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

{

if (lvSinhVien.SelectedItems.Count == 0)

{

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

return;

}

ListViewItem item = lvSinhVien.SelectedItems[0];

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

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

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

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

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

ClearTextBoxes();

}

// 🔴 Nút XÓA

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

{

if (lvSinhVien.SelectedItems.Count == 0)

{

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

return;

}

DialogResult result = MessageBox.Show("Bạn có chắc chắn muốn xóa sinh viên này?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (result == DialogResult.Yes)

{

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

ClearTextBoxes();

}

}

// ⚫ Nút THOÁT

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

{

DialogResult result = MessageBox.Show("Bạn có chắc muốn thoát không?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (result == DialogResult.Yes)

{

this.Close();

}

}

// 🟣 Khi chọn 1 dòng trong ListView

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

{

if (lvSinhVien.SelectedItems.Count > 0)

{

ListViewItem item = lvSinhVien.SelectedItems[0];

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

dtpNgaySinh.Value = DateTime.Parse(item.SubItems[1].Text);

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

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

}

}

// 🧹 Hàm xóa nội dung sau khi thêm/sửa/xóa

private void ClearTextBoxes()

{

txtHoTen.Clear();

txtLop.Clear();

txtDiaChi.Clear();

dtpNgaySinh.Value = DateTime.Now;

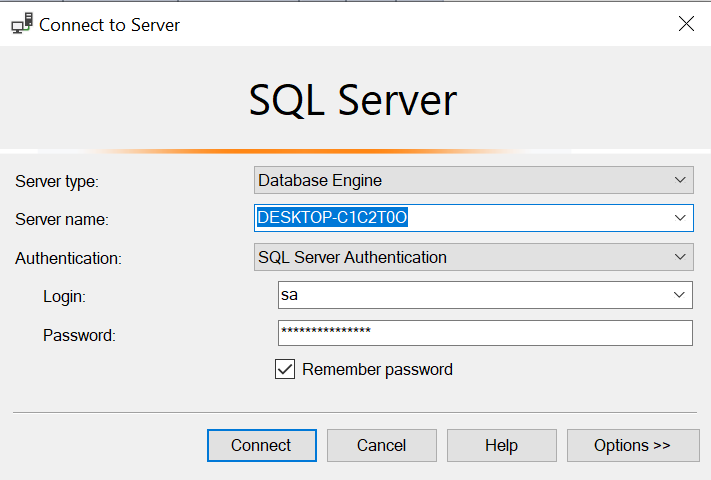
}

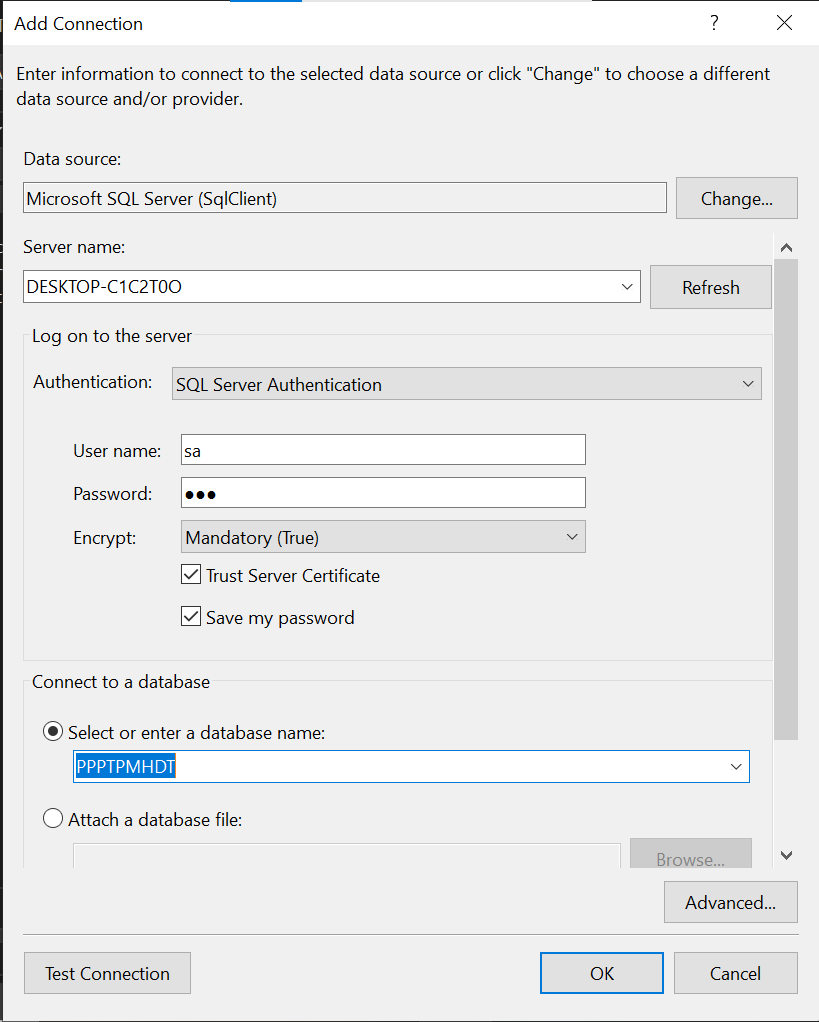
}

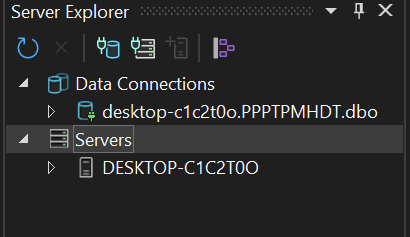
}

**Lab 4**

**Thực hành 1**







Chuỗi kết nối

Data Source=DESKTOP-C1C2T0O;Initial Catalog=PPPTPMHDT;Persist Security Info=True;User ID=sa;Password=\*\*\*\*\*\*\*\*\*\*\*;Trust Server Certificate=True

