Họ tên: Hoàng Trọng Dũng

Lớp: 11CNPM2

MSSV: 1150080129

Thực hành 1: Tạo và kết nối CSDL ngay bên trong Project Winform.

− Khởi tạo 1 project với template là Windowns Form Applications (.NET

framework).

− Chuột phải vào project chọn Add > New Item.

− Trong danh sách các template, cuộn xuống và chọn Service-based

Database.

A screenshot of a computer

AI-generated content may be incorrect.

using System;

using System.Data;

using System.Data.SqlClient;

using System.Windows.Forms;

namespace baitap\_oop

{

public partial class Form1 : Form

{

string strCon = @"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=""C:\Users\Dung\source\repos\baitap-oop\baitap-oop\DBConnect.mdf"";Integrated Security=True";

SqlConnection sqlCon = null;

public Form1()

{

InitializeComponent();

}

private void MoKetNoi()

{

if (sqlCon == null)

{

sqlCon = new SqlConnection(strCon);

}

if (sqlCon.State == ConnectionState.Closed)

{

sqlCon.Open();

}

}

private void DongKetNoi()

{

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

{

sqlCon.Close();

}

}

private void HienThiDanhSach()

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "SELECT MaSV, TenSV, GioiTinh, NgaySinh, QueQuan, MaLop FROM SinhVien";

sqlCmd.Connection = sqlCon;

lsvDanhSachSV.Items.Clear();

SqlDataReader reader = sqlCmd.ExecuteReader();

while (reader.Read())

{

string maSv = reader.IsDBNull(0) ? "" : reader.GetString(0);

string tenSV = reader.IsDBNull(1) ? "" : reader.GetString(1);

string gioiTinh = reader.IsDBNull(2) ? "" : reader.GetString(2);

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

string queQuan = reader.IsDBNull(4) ? "" : reader.GetString(4);

string maLop = reader.IsDBNull(5) ? "" : 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);

}

finally

{

DongKetNoi();

}

}

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

{

try

{

if (string.IsNullOrWhiteSpace(txtMaSV.Text) ||

string.IsNullOrWhiteSpace(txtTenSV.Text) ||

string.IsNullOrWhiteSpace(txtQueQuan.Text) ||

string.IsNullOrWhiteSpace(txtMaLop.Text) ||

cbGioiTinh.SelectedIndex == -1)

{

MessageBox.Show("Vui lòng điền đầy đủ thông tin!");

return;

}

MoKetNoi();

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();

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.Add("@MaSV", SqlDbType.NVarChar, 50).Value = maSV;

sqlCmd.Parameters.Add("@TenSV", SqlDbType.NVarChar, 100).Value = tenSV;

sqlCmd.Parameters.Add("@GioiTinh", SqlDbType.NVarChar, 10).Value = gioiTinh;

sqlCmd.Parameters.Add("@NgaySinh", SqlDbType.DateTime).Value = ngaySinh;

sqlCmd.Parameters.Add("@QueQuan", SqlDbType.NVarChar, 100).Value = queQuan;

sqlCmd.Parameters.Add("@MaLop", SqlDbType.NVarChar, 50).Value = maLop;

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

MessageBox.Show("Thêm sinh viên thành công!");

HienThiDanhSach();

txtMaSV.Clear();

txtTenSV.Clear();

txtQueQuan.Clear();

txtMaLop.Clear();

cbGioiTinh.SelectedIndex = -1;

dtpNgaySinh.Value = DateTime.Now;

}

}

catch (Exception ex)

{

MessageBox.Show("Thêm dữ liệu bị lỗi! " + ex.Message);

}

finally

{

DongKetNoi();

}

}

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

{

cbGioiTinh.Items.Add("Nam");

cbGioiTinh.Items.Add("Nữ");

this.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F, System.Drawing.FontStyle.Regular);

HienThiDanhSach();

}

}

}

Thực hành 2: Thêm dữ liệu không dùng Parameter.

− Sử dụng hàm ExecuteNonQuery để thực hiện thêm dữ liệu.

− Ví dụ: Thêm dữ liệu cho 1 sinh viên vào bảng SinhVien.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

using System;

using System.Data;

using System.Data.SqlClient;

using System.Windows.Forms;

using System.Threading;

namespace baitap\_oop

{

public partial class Form2 : Form

{

string strCon = @"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=""C:\Users\Dung\source\repos\baitap-oop\baitap-oop\DBConnect.mdf"";Integrated Security=True;MultipleActiveResultSets=True;App=EntityFramework";

SqlConnection sqlCon = null;

public Form2()

{

InitializeComponent();

Thread.CurrentThread.CurrentCulture = new System.Globalization.CultureInfo("vi-VN");

Thread.CurrentThread.CurrentUICulture = new System.Globalization.CultureInfo("vi-VN");

}

private void MoKetNoi()

{

if (sqlCon == null)

{

sqlCon = new SqlConnection(strCon);

}

if (sqlCon.State == ConnectionState.Closed)

{

sqlCon.Open();

}

}

private void DongKetNoi()

{

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

{

sqlCon.Close();

}

}

private void LoadMaLop()

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "SELECT DISTINCT MaLop FROM SinhVien WHERE MaLop IS NOT NULL ORDER BY MaLop";

sqlCmd.Connection = sqlCon;

SqlDataReader reader = sqlCmd.ExecuteReader();

cbMaLop.Items.Clear();

while (reader.Read())

{

if (!reader.IsDBNull(0))

{

string maLop = reader.GetString(0);

if (!string.IsNullOrWhiteSpace(maLop))

{

cbMaLop.Items.Add(maLop);

}

}

}

reader.Close();

}

catch (Exception ex)

{

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

}

finally

{

DongKetNoi();

}

}

private void HienThiSinhVienTheoLop(string maLop)

{

try

{

MoKetNoi();

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

sqlCmd.CommandText = "SELECT MaSV, TenSV, GioiTinh, NgaySinh, QueQuan, MaLop FROM SinhVien WHERE MaLop = @MaLop ORDER BY MaSV";

sqlCmd.Connection = sqlCon;

sqlCmd.Parameters.Add("@MaLop", SqlDbType.NVarChar, 50).Value = maLop;

lsvDanhSach.Items.Clear();

SqlDataReader reader = sqlCmd.ExecuteReader();

while (reader.Read())

{

string maSv = reader.IsDBNull(0) ? "" : reader.GetString(0);

string tenSV = reader.IsDBNull(1) ? "" : reader.GetString(1);

string gioiTinh = reader.IsDBNull(2) ? "" : reader.GetString(2);

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

string queQuan = reader.IsDBNull(4) ? "" : reader.GetString(4);

string maLopSV = reader.IsDBNull(5) ? "" : 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);

}

finally

{

DongKetNoi();

}

}

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

{

if (cbMaLop.SelectedIndex != -1)

{

string maLop = cbMaLop.SelectedItem.ToString();

HienThiSinhVienTheoLop(maLop);

}

}

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

{

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

try

{

ListViewItem lvi = lsvDanhSach.SelectedItems[0];

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

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

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

if (DateTime.TryParse(lvi.SubItems[3].Text, out DateTime ngaySinh))

{

dtpNgaySinh.Value = ngaySinh;

}

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

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

}

catch (Exception ex)

{

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

}

}

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

{

try

{

if (string.IsNullOrWhiteSpace(txtMaSV.Text) ||

string.IsNullOrWhiteSpace(txtTenSV.Text) ||

string.IsNullOrWhiteSpace(txtQueQuan.Text) ||

string.IsNullOrWhiteSpace(txtMaLop.Text) ||

cbGioiTinh.SelectedIndex == -1)

{

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

return;

}

MoKetNoi();

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();

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;

sqlCmd.Parameters.Add("@MaSV", SqlDbType.NVarChar, 50).Value = maSV;

sqlCmd.Parameters.Add("@TenSV", SqlDbType.NVarChar, 100).Value = tenSV;

sqlCmd.Parameters.Add("@GioiTinh", SqlDbType.NVarChar, 10).Value = gioiTinh;

sqlCmd.Parameters.Add("@NgaySinh", SqlDbType.DateTime).Value = ngaySinh;

sqlCmd.Parameters.Add("@QueQuan", SqlDbType.NVarChar, 100).Value = queQuan;

sqlCmd.Parameters.Add("@MaLop", SqlDbType.NVarChar, 50).Value = maLop;

int kq = sqlCmd.ExecuteNonQuery();

if (kq > 0)

{

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

if (cbMaLop.SelectedIndex != -1)

{

HienThiSinhVienTheoLop(cbMaLop.SelectedItem.ToString());

}

}

else

{

MessageBox.Show("Cập nhật không thành công - Không tìm thấy sinh viên!", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

catch (Exception ex)

{

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

}

finally

{

DongKetNoi();

}

}

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

{

cbGioiTinh.Items.Add("Nam");

cbGioiTinh.Items.Add("Nữ");

this.Font = new System.Drawing.Font("Segoe UI", 9F, System.Drawing.FontStyle.Regular);

lsvDanhSach.Font = new System.Drawing.Font("Segoe UI", 9F, System.Drawing.FontStyle.Regular);

txtMaSV.Font = new System.Drawing.Font("Segoe UI", 9F, System.Drawing.FontStyle.Regular);

txtTenSV.Font = new System.Drawing.Font("Segoe UI", 9F, System.Drawing.FontStyle.Regular);

txtQueQuan.Font = new System.Drawing.Font("Segoe UI", 9F, System.Drawing.FontStyle.Regular);

txtMaLop.Font = new System.Drawing.Font("Segoe UI", 9F, System.Drawing.FontStyle.Regular);

cbGioiTinh.Font = new System.Drawing.Font("Segoe UI", 9F, System.Drawing.FontStyle.Regular);

cbMaLop.Font = new System.Drawing.Font("Segoe UI", 9F, System.Drawing.FontStyle.Regular);

LoadMaLop();

}

}

}

Thực hành 3: Sửa dữ liệu

− Sử dụng hàm ExecuteNonQuery để thực hiện sửa dữ liệu.

− Ví dụ: Sửa dữ liệu cho 1 sinh viên trong bảng SinhVien.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Thực hành 4: Xóa dữ liệu không dùng Parameter.

− Sử dụng hàm ExecuteNonQuery để thực hiện xóa dữ liệu.

− Ví dụ: xóa dữ liệu của 1 sinh viên trong bảng SinhVien.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

using System;

using System.Data;

using System.Data.SqlClient;

using System.Windows.Forms;

namespace baitap\_oop

{

public partial class Form3 : Form

{

string strCon = @"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=""C:\Users\Dung\source\repos\baitap-oop\baitap-oop\DBConnect.mdf"";Integrated Security=True";

SqlConnection sqlCon = null;

string maSV = "";

public Form3()

{

InitializeComponent();

}

private void MoKetNoi()

{

if (sqlCon == null)

{

sqlCon = new SqlConnection(strCon);

}

if (sqlCon.State == ConnectionState.Closed)

{

sqlCon.Open();

}

}

private void DongKetNoi()

{

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

{

sqlCon.Close();

}

}

private void HienThiDanhSach()

{

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();

}

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

{

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

ListViewItem lvi = lsvDanhSach.SelectedItems[0];

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

}

private void XoaSV(string maSV)

{

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 dữ liệu thành công!");

HienThiDanhSach();

}

else

{

MessageBox.Show("Xóa không thành công!");

}

}

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.Error);

}

else

{

DialogResult result = MessageBox.Show(

"Bạn có thực sự muốn xóa sinh viên " + maSV + " không?",

"Xác nhận xóa",

MessageBoxButtons.YesNo,

MessageBoxIcon.Warning);

if (result == DialogResult.Yes)

{

XoaSV(maSV);

maSV = "";

}

}

}

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

{

HienThiDanhSach();

}

}

}