KIỂM TRA CUỐI KỲ

Môn: ADO.NET

**Tên các thành viên:**

Nguyễn Vũ Phương Loan-162000244

Trần Thị Thúy Vi-162000128

**ĐỀ 1**

Xây dựng chương trình quản lý nhân viên cho một công ty. Thông tin nhân viên bao gồm, mã nhân viên, họ tên, ngày tháng năm sinh, giới tính, trình độ, loại nhân viên (văn phòng, công nhân, quản lý), lương hàng tháng.

Yêu cầu:

- Xây dựng chức năng quản lý tài khoản, chỉ có tài khoản enable được đăng nhập)

(tạo form để nhập tài khoản và cho phép tài khoản nào được đăng nhập)

- Xây dựng Form quản lý nhân viên với chức năng:

+ Thêm mới nhân viên sẽ thêm 1 dòng trong lưới danh sách nhân viên.

+ Xóa nội dung khi chọn 1 dòng trên lưới, thông tin chi tiết hiện trên màn hình phía trên

 + Sửa nội dung khi chọn 1 dòng trên lưới, thông tin chi tiết hiện trên màn hình phía trên

- Tạo form báo cáo thông tin nhân viên theo loại nhân viên, giới tính.

- Thiết kế mô hình 3 lớp

- Chức năng sao lưu và phục hồi dữ liệu (khi bị lỗi hay mất dữ liệu, ví dụ: xóa nhầm, sửa nhầm)

**Bài làm**

**Tạo một Form Đăng nhập:**

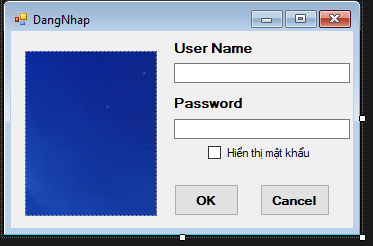
Gồm 3 label: Username, Password và một label trống để hiển thị thông báo khi nhập sai mật khẩu.

2 textbox để nhập tên và mật khẩu

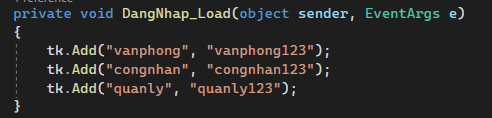
1 checkbox để hiển thị mật khẩu

2 button để đăng nhập và đăng xuất

1 pictureBox để hiện hình ảnh

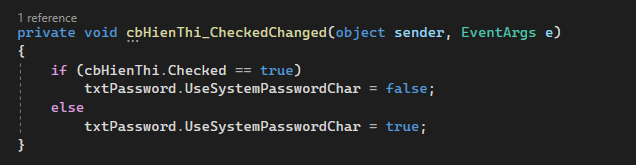


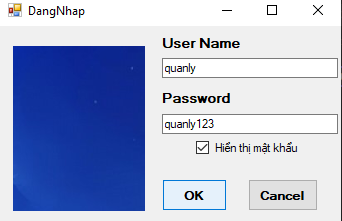
Tạo các tài khoản dùng để đăng nhập



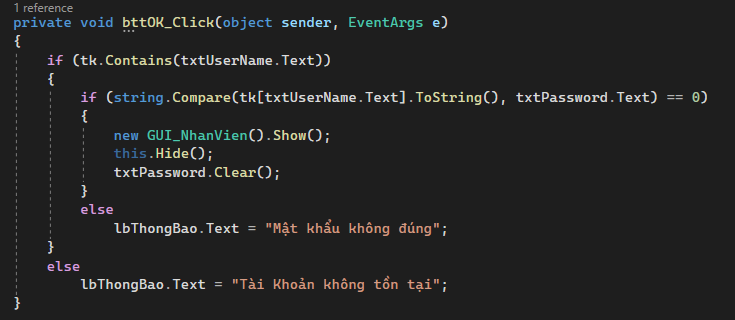
Sử dụng checkbox để hiển thị mật khẩu

Sử dụng CheckedChanged để hiển thị mật khẩu

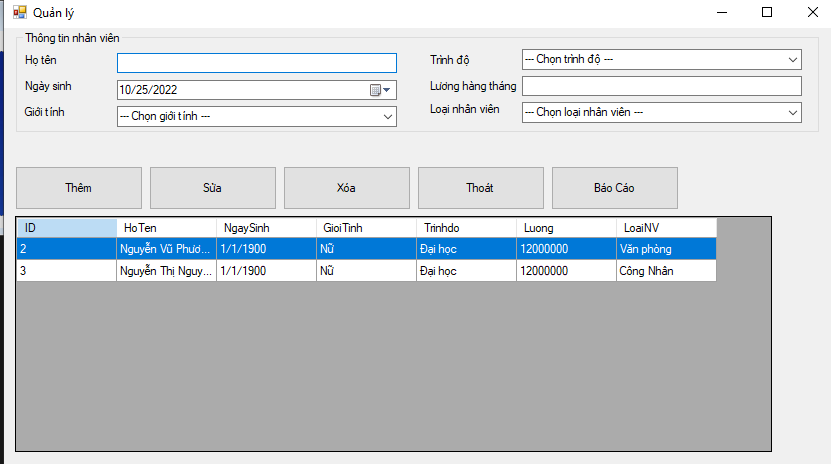




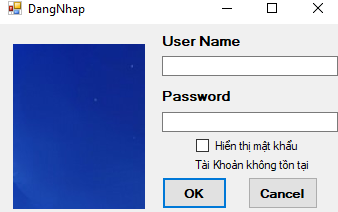
Tạo button Ok để đăng nhập sau khi đã nhập UserName và Password



Sau khi bấm nút đăng nhập tài khoản đúng sẽ chuyển sang trang quản lý



Nhập sai thì sẽ yêu cầu nhập lại



**Tiếp theo tạo một Form Quản lý Nhân viên**

Gồm:

6 label: Họ tên, Ngày sinh, giới tính, trình độ, lương, loại nhân viên.

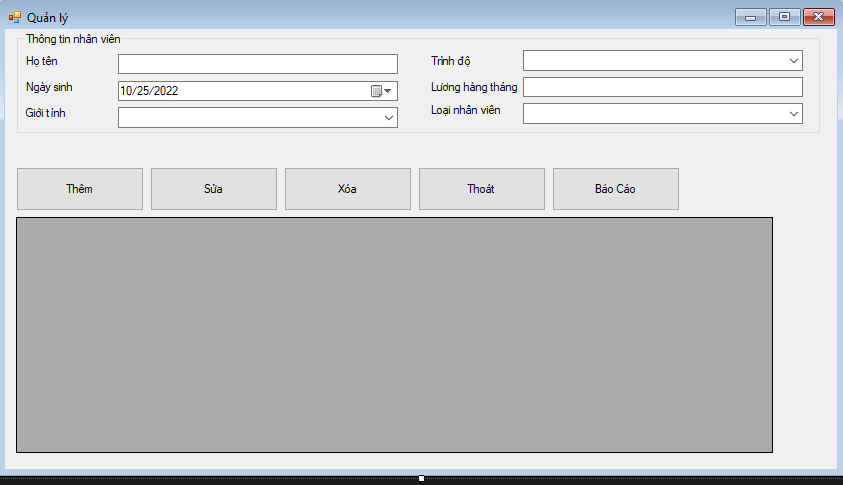
5 button: Thêm, sửa, xóa, thoát và báo cáo

3 combobox cho giới tính, trình độ , loại nhân viên

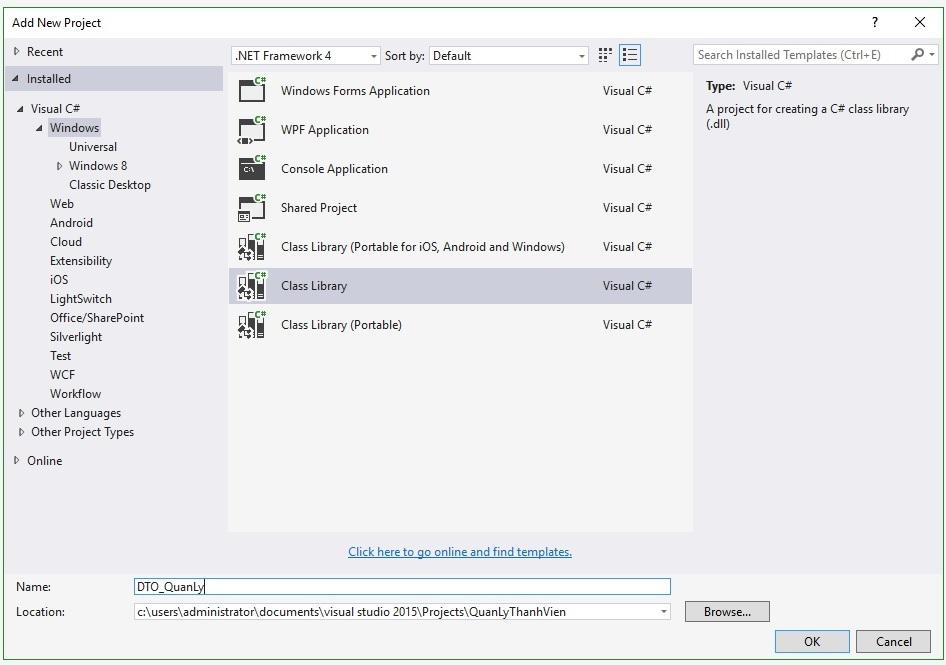
2 textbox cho Họ tên, lương

1 DateTimePicker cho ngày sinh

Căn chỉnh phù hợp



Bây giờ chúng ta bắt đầu tạo, đối với 3 Project DTO, Business và Data Access chúng ta tạo theo **Class Library**



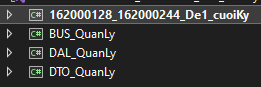
Khi tạo xong chúng ta sẽ được các lớp như sau

Lớp BUS\_QuanLy

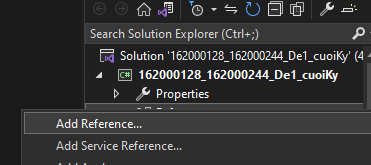
Lớp DAL\_QuanLy

Lớp DTO\_QuanLy

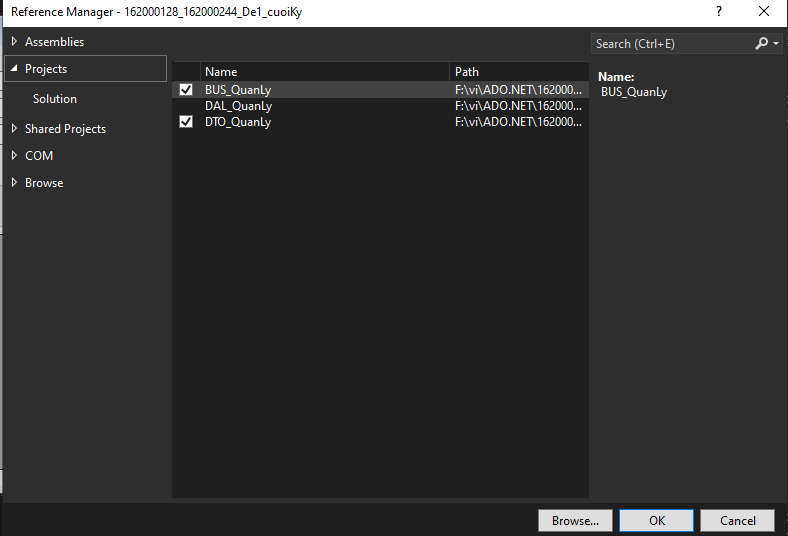
Lớp GUI là 162000128\_162000244\_De1\_cuoiKy



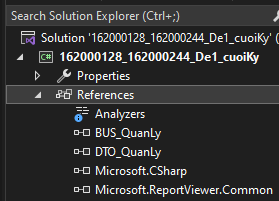
Tiếp theo sẽ liên kết, làm project GUI. Đầu tiên ta chọn chuột phải vào **References => Add Reference**



Một cái bảng hiện ra, tại mục Project ta sẽ chọn 2 lớp mà lớp GUI có thể liên kết tới là **BUS** và **DTO**:

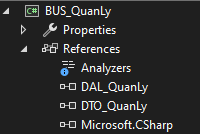


Kết quả

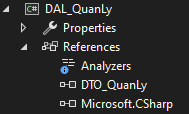


Làm tương tự với lớp BUS và DAL

Kết quả lớp BUS



Lớp DAL



**Sử dụng cơ sở dữ liệu như sau:**

CREATE TABLE Nhan\_Vien1 (

ID INT NOT NULL PRIMARY KEY IDENTITY,

HoTen nvarchar(50) NOT NULL,

NgaySinh datetime NOT NULL,

GioiTinh nvarchar(10) NOT NULL,

Trinhdo nvarchar(50) NOT NULL,

Luong int NOT NULL,

LoaiNV nvarchar(20) NOT NULL,

)

SET IDENTITY\_INSERT [dbo].[Nhan\_Vien1] ON

INSERT [dbo].[Nhan\_Vien1] ([Id], [HoTen], [NgaySinh], [GioiTinh], [TrinhDo], [Luong], [LoaiNV]) VALUES (1, N'Trần Thị Thúy Vi', 6/2/2000, N'Nữ', N'Đại học', 12000000, N'Quản lý')

INSERT [dbo].[Nhan\_Vien1] ([Id], [HoTen], [NgaySinh], [GioiTinh], [TrinhDo], [Luong], [LoaiNV]) VALUES (2,N'Nguyễn Vũ Phương Loan', 6/10/2000, N'Nữ', N'Đại học', 12000000, N'Văn phòng')

INSERT [dbo].[Nhan\_Vien1] ([Id], [HoTen], [NgaySinh], [GioiTinh], [TrinhDo], [Luong], [LoaiNV]) VALUES (3, N'Nguyễn Thị Nguyệt Quế', 6/9/2000, N'Nữ', N'Đại học', 12000000, N'Công Nhân')

SET IDENTITY\_INSERT [dbo].[Nhan\_Vien1] OFF

**Xây dựng lớp DTO**

Mình sẽ tạo file **DTO\_NhanVien**.cs (Class file), **DTO\_NhanVien**.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DTO\_QuanLy

{

public class DTO\_NhanVien

{

private int \_NHANVIEN\_ID;

private string \_NHANVIEN\_HOTEN;

private string \_NHANVIEN\_NGAYSINH;

private string \_NHANVIEN\_GIOITINH;

private string \_NHANVIEN\_TRINHDO;

private string \_NHANVIEN\_LUONGTHANG;

private string \_NHANVIEN\_LOAINV;

public int NHANVIEN\_ID

{

get

{

return \_NHANVIEN\_ID;

}

set

{

\_NHANVIEN\_ID = value;

}

}

public string NHANVIEN\_HOTEN

{

get

{

return \_NHANVIEN\_HOTEN;

}

set

{

\_NHANVIEN\_HOTEN = value;

}

}

public string NHANVIEN\_NGAYSINH

{

get

{

return \_NHANVIEN\_NGAYSINH;

}

set

{

\_NHANVIEN\_NGAYSINH = value;

}

}

public String NHANVIEN\_GIOITINH

{

get

{

return \_NHANVIEN\_GIOITINH;

}

set

{

\_NHANVIEN\_GIOITINH = value;

}

}

public String NHANVIEN\_TRINHDO

{

get

{

return \_NHANVIEN\_TRINHDO;

}

set

{

\_NHANVIEN\_TRINHDO = value;

}

}

public string NHANVIEN\_LUONGTHANG

{

get

{

return \_NHANVIEN\_LUONGTHANG;

}

set

{

\_NHANVIEN\_LUONGTHANG = value;

}

}

public string NHANVIEN\_LOAINV

{

get

{

return \_NHANVIEN\_LOAINV;

}

set

{

\_NHANVIEN\_LOAINV = value;

}

}

public DTO\_NhanVien()

{

}

public DTO\_NhanVien(int ID, string hoten, string ngaysinh,string gioitinh, string trinhdo, string luongthang, string loainv)

{

this.NHANVIEN\_ID = ID;

this.NHANVIEN\_HOTEN = hoten;

this.NHANVIEN\_NGAYSINH = ngaysinh;

this.NHANVIEN\_GIOITINH = gioitinh;

this.NHANVIEN\_TRINHDO = trinhdo;

this.NHANVIEN\_LUONGTHANG = luongthang;

this.NHANVIEN\_LOAINV = loainv;

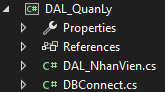
}

}

}

**Tạo lớp DAL**

Tạo 2 lớp DAL\_NhanVien và DBConnect trong lớpDAL\_QuanLy



Cần tạo class **DBConnect.cs** với nội dung như sau:

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DAL\_QuanLy

{

public class DBConnect

{

protected SqlConnection \_conn = new SqlConnection("Data Source=DESKTOP-BDUR7FC\\SQLEXPRESS;Initial Catalog=QuanLyNhanVien;Integrated Security=True");

}

}

Tạo **SqlConnection** và khởi tạo luôn, sau này các**class DAL** chúng ta chỉ cần **kế thừa class DBConnect** là có thể sử dụng **\_conn** thoải mái không cần khởi tạo lại.

Tạo file **DAL\_NhanVien**.cs (Class file) kế thừa lớp DBConnect

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using DTO\_QuanLy;

namespace DAL\_QuanLy

{

public class DAL\_NhanVien : DBConnect

{

public DataTable getNhanVien()

{

SqlDataAdapter da = new SqlDataAdapter("SELECT \* FROM NHAN\_VIEN1", \_conn);

DataTable dtThanhvien = new DataTable();

da.Fill(dtThanhvien);

return dtThanhvien;

}

public bool themNhanVien(DTO\_NhanVien nv)

{

try

{

\_conn.Open();

string SQL = string.Format("INSERT INTO NHAN\_VIEN1(Hoten, NgaySinh, GioiTinh, TrinhDo, Luong, LoaiNV) VALUES ('{0}', '{1}', '{2}', '{3}', '{4}', '{5}')", nv.NHANVIEN\_HOTEN, nv.NHANVIEN\_NGAYSINH, nv.NHANVIEN\_GIOITINH, nv.NHANVIEN\_TRINHDO, nv.NHANVIEN\_LUONGTHANG, nv.NHANVIEN\_LOAINV);

SqlCommand cmd = new SqlCommand(SQL, \_conn);

if (cmd.ExecuteNonQuery() > 0)

return true;

}

catch (Exception e)

{

}

finally

{

\_conn.Close();

}

return false;

}

public bool suaNhanVien(DTO\_NhanVien nv )

{

try

{

\_conn.Open();

string SQL = string.Format("UPDATE NHAN\_VIEN1 SET HoTen = '{0}' , NgaySinh = '{1}', GioiTinh = '{2}', TrinhDo = '{3}', Luong = '{4}', LoaiNV = '{5}' where ID = '{6}'", nv.NHANVIEN\_HOTEN, nv.NHANVIEN\_NGAYSINH, nv.NHANVIEN\_GIOITINH, nv.NHANVIEN\_TRINHDO, nv.NHANVIEN\_LUONGTHANG, nv.NHANVIEN\_LOAINV);

SqlCommand cmd = new SqlCommand(SQL, \_conn);

if (cmd.ExecuteNonQuery() > 0)

return true;

}

catch (Exception e)

{

}

finally

{

\_conn.Close();

}

return false;

}

public bool xoaNhanVien(int ID)

{

try

{

\_conn.Open();

string SQL = string.Format("DELETE FROM NHAN\_VIEN1 WHERE ID ='{0}'", ID);

SqlCommand cmd = new SqlCommand(SQL, \_conn);

if (cmd.ExecuteNonQuery() > 0)

return true;

}

catch (Exception e)

{

}

finally

{

\_conn.Close();

}

return false;

}

}

}

**Tạo lớp BUS**

Bước này là bước xử lý business logic (Business layer).

Ở bước này, ta có thể lấy dữ liệu từ **DAL** về, xử lý ABC XYZ gì đó rồi trả về lại cho **GUI** sử dụng. Hoặc khi update dữ liệu trên DB, **GUI** gửi data lên **BUS** và rồi ta xử lý ABC XYZ gì đó cho data của chúng ta, rồi mới insert/update/delete chẳng hạn,…

**BUS\_NhanVien**.cs (Class file):

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using DAL\_QuanLy;

using DTO\_QuanLy;

namespace BUS\_QuanLy

{

public class BUS\_NhanVien

{

DAL\_NhanVien dalNhanvien = new DAL\_NhanVien();

public DataTable getNhanVien()

{

return dalNhanvien.getNhanVien();

}

public bool themNhanVien(DTO\_NhanVien nv)

{

return dalNhanvien.themNhanVien(nv);

}

public bool suaNhanVien(DTO\_NhanVien nv)

{

return dalNhanvien.suaNhanVien(nv);

}

public bool xoaNhanVien(int ID)

{

return dalNhanvien.xoaNhanVien(ID);

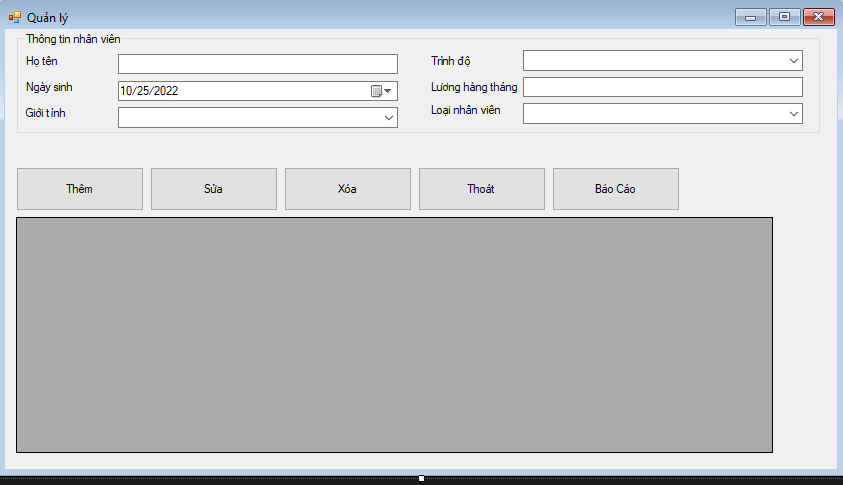
}

}

}

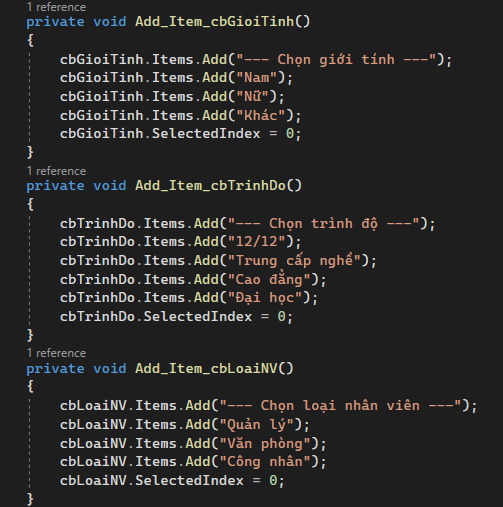
**Tạo Lớp GUI**

Design form



Lần lượt gán các chức năng vào

Tạo các combobox để lựa chọn



Chức năng thêm Nhân viên:

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

{

if (txtName.Text != "" && dtBirth.Text != "" && cbGioiTinh.Text != "" && cbTrinhDo.Text != "" && txtLuong.Text != "" && cbLoaiNV.Text != "")

{

DTO\_NhanVien tv = new DTO\_NhanVien(0, txtName.Text, dtBirth.Text, cbGioiTinh.Text, cbTrinhDo.Text, txtLuong.Text, cbLoaiNV.Text);

if (busNV.themNhanVien(tv))

{

MessageBox.Show("Thêm thành công");

dgvNV.DataSource = busNV.getNhanVien();

}

else

{

MessageBox.Show("Thêm không thành công");

}

}

else

{

MessageBox.Show("Xin hãy nhập đầy đủ");

}

}

Chức năng Sửa Nhân viên:

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

{

if (dgvNV.SelectedRows.Count > 0)

{

if (txtName.Text != "" && dtBirth.Text != "" && cbGioiTinh.Text != "" && cbTrinhDo.Text != "" && txtLuong.Text != "" && cbLoaiNV.Text != "")

{

DataGridViewRow row = dgvNV.SelectedRows[0];

int ID = Convert.ToInt16(row.Cells[0].Value.ToString());

DTO\_NhanVien tv = new DTO\_NhanVien(ID, txtName.Text, dtBirth.Text, cbGioiTinh.Text, cbTrinhDo.Text, txtLuong.Text, cbLoaiNV.Text);

if (busNV.suaNhanVien(tv))

{

MessageBox.Show("Sửa thành công");

dgvNV.DataSource = busNV.getNhanVien();

}

else

{

MessageBox.Show("Sửa không thành công");

}

}

else

{

MessageBox.Show("Xin hãy nhập đầy đủ");

}

}

}

Chức năng xóa Nhân viên:

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

{

if (dgvNV.SelectedRows.Count > 0)

{

DataGridViewRow row = dgvNV.SelectedRows[0];

int ID = Convert.ToInt16(row.Cells[0].Value.ToString());

if (busNV.xoaNhanVien(ID))

{

MessageBox.Show("Xóa thành công");

dgvNV.DataSource = busNV.getNhanVien();

}

else

{

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

}

}

else

{

MessageBox.Show("Hãy chọn thành viên muốn xóa");

}

}

Chức năng thoát

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

{

DialogResult = MessageBox.Show("Bạn có chắc chắn muốn thoát?", "Thông báo", MessageBoxButtons.OKCancel, MessageBoxIcon.Warning);

if (DialogResult == DialogResult.OK)

{

Application.Exit();

}

}

Chức năng Load để hiển thị trên form:

private void GUI\_NhanVien\_Load(object sender, EventArgs e)

{

Add\_Item\_cbLoaiNV();

Add\_Item\_cbTrinhDo();

Add\_Item\_cbGioiTinh();

dgvNV.DataSource = busNV.getNhanVien();

dgvNV.SelectionMode = DataGridViewSelectionMode.FullRowSelect;

}

Chức năng khi Click vào lưới thì hiển thị lên các dữ kiện:

private void dgvNV\_CellClick(object sender, DataGridViewCellEventArgs e)

{

// Lấy row hiện tại

DataGridViewRow row = dgvNV.SelectedRows[0];

// Chuyển giá trị lên form

txtName.Text = row.Cells[1].Value.ToString();

dtBirth.Text = row.Cells[2].Value.ToString();

cbGioiTinh.Text = row.Cells[3].Value.ToString();

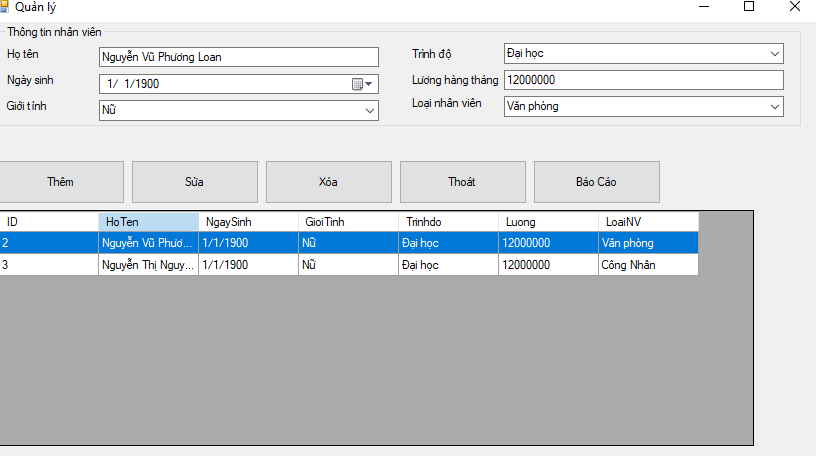
cbTrinhDo.Text = row.Cells[4].Value.ToString();

txtLuong.Text = row.Cells[5].Value.ToString();

cbLoaiNV.Text = row.Cells[6].Value.ToString();

}

Kết quả:



Chức năng báo cáo:

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

{

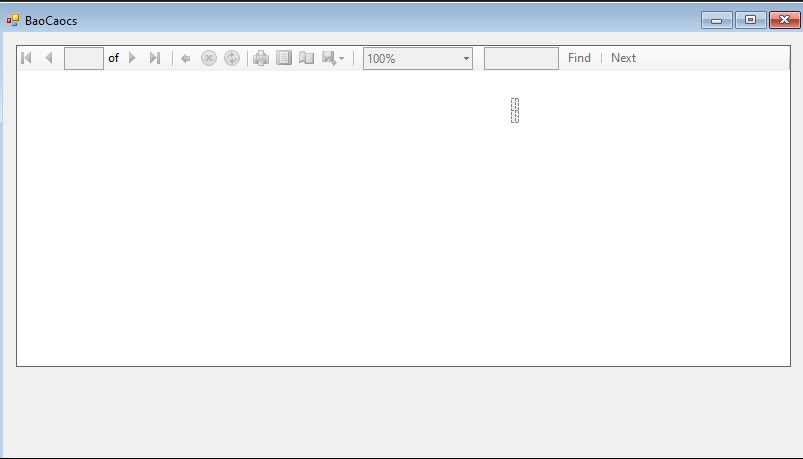
new BaoCaocs().Show();

}

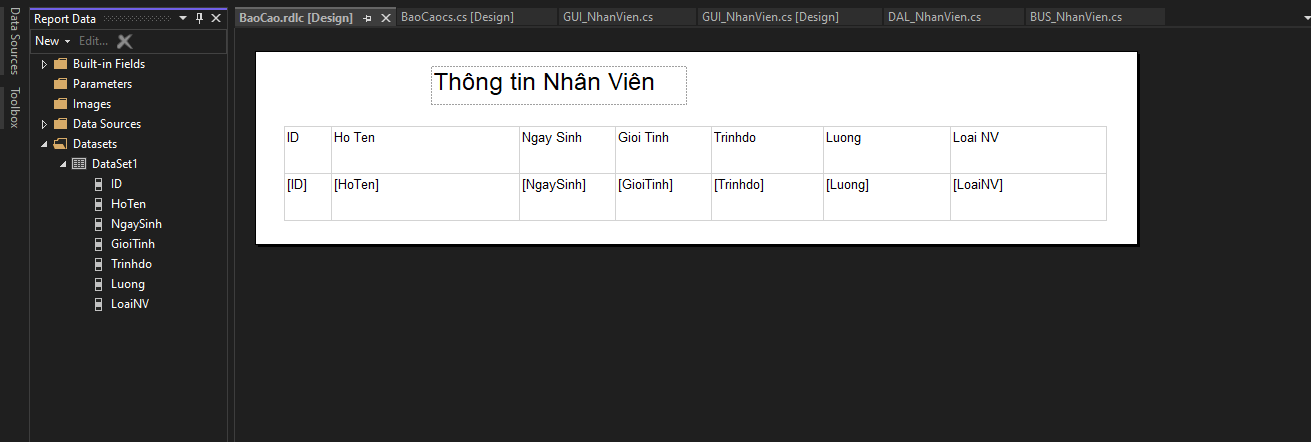
**Tạo Form Report:**

Vào toolbox chọn ReportViewer





Tạo Dataset:



Kết quả:

