**Lab 04**

**Truy cập dữ liệu sử dụng ADO.NET**

**Phần I: Bài tập step by step**

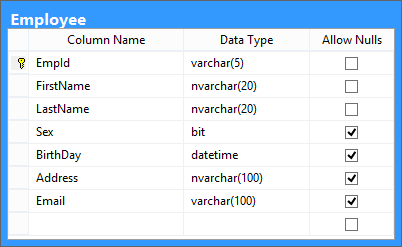
**Mục tiêu:**

* Kết nối với cơ sở dữ liệu
* Thực hiện các thao tác với cơ sở dữ liệu

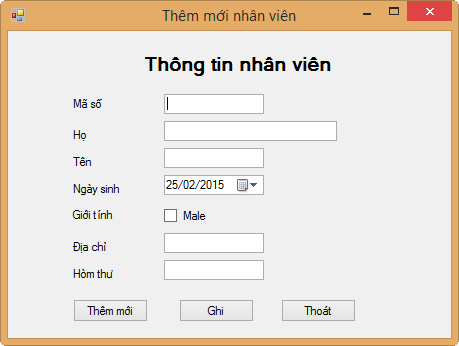
**Bài 4.1**

**Tạo ứng dụng Windows Form cho phép lưu trữ thông tin nhân viên vào cơ sở dữ liệu**

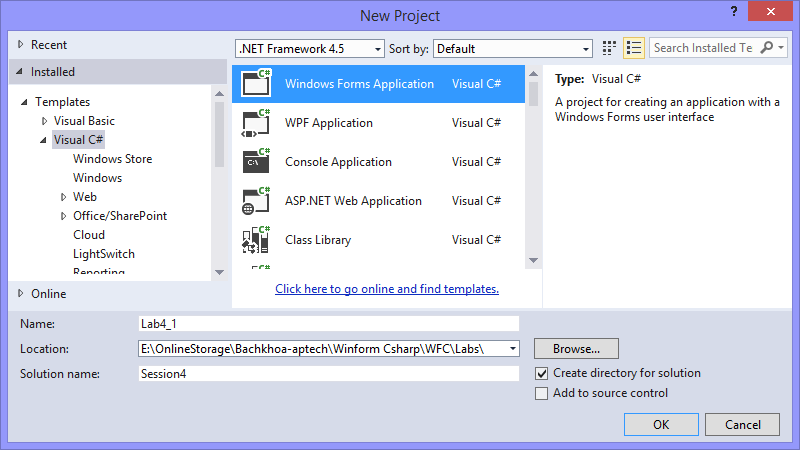
* *Cho cơ sở dữ liệu LabWinform với bảng có cấu trúc như sau*



* *Giao diện màn hình thêm nhân viên*



**Bước 1:** Mở Visual Studio 2013 -> Click menu File -> New -> Project -> chọn loại Windows Forms Appliaction -> Nhập tên Project -> Tên Solution như hình dưới -> OK



**Bước 2:** Thiết kế form theo đầu bài yêu cầu và đặt tên cho các control theo bảng sau (chú ý các control label không cần đặt tên):

|  |  |  |
| --- | --- | --- |
| **Loại Control** | **Đặt tên** | **Thuộc tính khác** |
| Màn hình: Form | frmEmployee | Text:Thêm mới nhân viên |
| Mã số: TextBox | txtId |  |
| Họ: TextBox | txtFirstName |  |
| Tên: TextBox | txtLastName |  |
| Ngày sinh: DateTimePicker | txtBirthday | Format:Custom  CustomFormat: dd/MM/yyyy |
| Giới tính: CheckBox | chkSex | Text: Male |
| Địa chỉ: TextBox | txtAddress |  |
| Hòm thư: TextBox | txtEmail |  |
| Thêm mới: Button | btnClear | Text: Thêm mới |
| Ghi: Button | btnSave | Text: Ghi |
| Thoát: Button | btnExit | Text: Thoát |

**Bước 3:** Code cho chương trình:

* *Đưa thư viện vào chương trình*

using System.Data.SqlClient;

* *Khai báo biến*

//khai báo chuỗi kết nối

string sqlconnect = "server=CHUNGLD\\SQLEXPRESS;database=LabWinform;uid=sa;pwd=123465";

//khai báo đối tượng kết nối

SqlConnection con;

* *Sự kiện Load của frmEmployee*

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

{

//khởi tạo và mở connection

con = new SqlConnection(sqlconnect);

con.Open();

}

* *Sự kiện Click của nút Ghi*

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

{

try

{

//tạo đối tượng command

SqlCommand cmd = con.CreateCommand();

//thiết lập câu lệnh insert

cmd.CommandText = "insert into Employee values(@EmpId,@FirstName," +

"@LastName,@Sex,@Birthday,@Address,@Email)";

//tạo tham số và gán giá trị cho tham số

cmd.Parameters.Add("@EmpId", SqlDbType.VarChar, 5).Value = txtId.Text;

cmd.Parameters.Add("@FirstName", SqlDbType.NVarChar, 20).Value = txtFirstName.Text;

cmd.Parameters.Add("@LastName", SqlDbType.NVarChar, 20).Value = txtLastName.Text;

cmd.Parameters.Add("@Sex", SqlDbType.Bit).Value = chkSex.Checked;

cmd.Parameters.Add("@Birthday", SqlDbType.DateTime).Value = txtBirthday.Value;

cmd.Parameters.Add("@Address", SqlDbType.NVarChar, 100).Value = txtAddress.Text;

cmd.Parameters.Add("@Email", SqlDbType.NVarChar, 100).Value = txtEmail.Text;

//thực thi câu truy vấn

cmd.ExecuteNonQuery();

MessageBox.Show("Thêm thành công!", "Thêm mới");

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message, "Thêm mới", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

* *Sự kiện Click của nút Thêm mới*

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

{

//xóa trắng form

txtId.Text = txtFirstName.Text = txtLastName.Text = txtEmail.Text = txtAddress.Text = "";

chkSex.Checked = false;

}

* *Sự kiện Click của nút Exit*

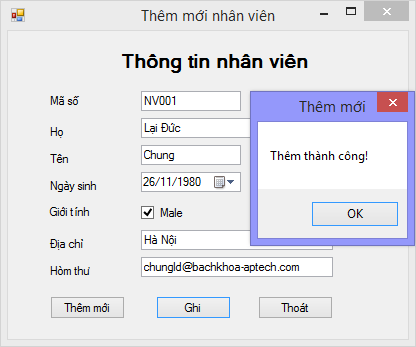
private void btnExit\_Click(object sender, EventArgs e)

{

this.Close();

}

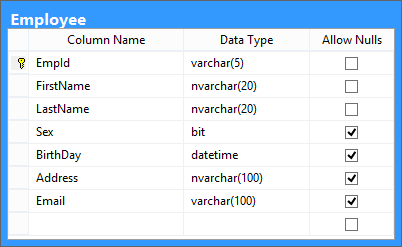
**Bước 4:** F5 để chạy và kiểm thử



**Bài 4.2**

**Tạo ứng dụng Windows Form cho phép đọc, xóa, sửa thông tin nhân viên dựa vào mã số**

* *Cho cơ sở dữ liệu LabWinform với bảng cơ cấu trúc như sau*



* *Cho thủ tục cập nhật nhân viên như sau*

Create proc [UpdateEmployee]

@EmpId varchar(5),

@FirstName nvarchar(20),

@LastName nvarchar(20),

@Sex bit,

@BirthDay datetime,

@Address nvarchar(100),

@Email varchar(100)

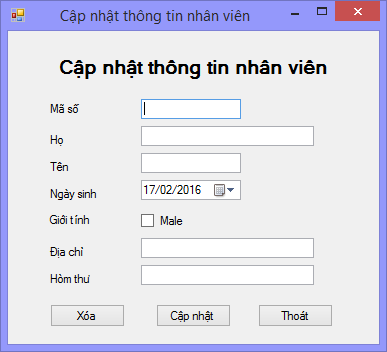
as

update employee set empid=@EmpId,firstname=@FirstName,

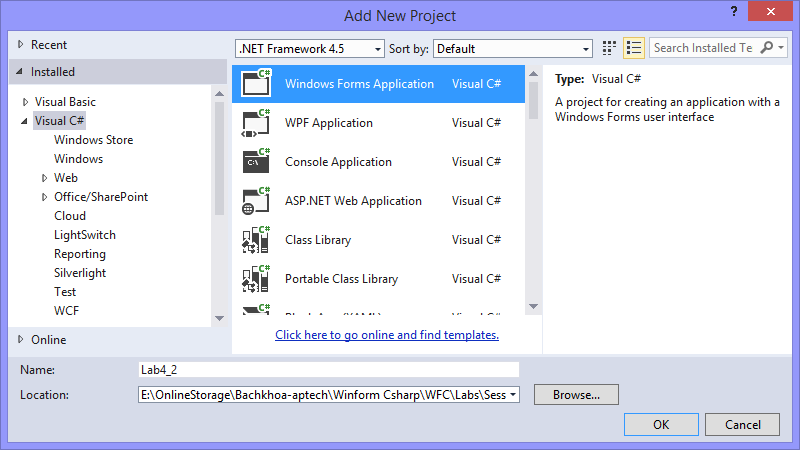
lastname=@LastName,sex=@Sex,birthday=@BirthDay,address=@Address,

email=@Email where empid=@empid

* *Giao diện màn hình cập nhật thông tin nhân viên*



**Bước 1:** Kích chuột phải vào Solution “Session4” -> Add -> New Project.. -> đặt tên như hình



**Bước 2:** Thiết kế form theo đầu bài yêu cầu và đặt tên cho các control theo bảng sau (chú ý các control label không cần đặt tên):

|  |  |  |
| --- | --- | --- |
| **Loại Control** | **Đặt tên** | **Thuộc tính khác** |
| Màn hình: Form | frmEditEmployee | Text:Cập nhật thông tin nhân viên |
| Mã số: TextBox | txtId |  |
| Họ: TextBox | txtFirstName |  |
| Tên: TextBox | txtLastName |  |
| Ngày sinh: DateTimePicker | txtBirthday | Format:Custom  CustomFormat: dd/MM/yyyy |
| Giới tính: CheckBox | chkSex | Text: Male |
| Địa chỉ: TextBox | txtAddress |  |
| Hòm thư: TextBox | txtEmail |  |
| Xóa: Button | btnDelete | Text: Xóa |
| Cập nhật: Button | btnUpdate | Text:Cập nhật |
| Thoát: Button | btnExit | Text: Thoát |

**Bước 3:** Code cho chương trình:

* *Đưa thư viện vào chương trình*

using System.Data.SqlClient;

* *Khai báo biến*

//khai báo chuỗi kết nối

string sqlconnect = "server=CHUNGLD\\SQLEXPRESS;database=LabWinform;uid=sa;pwd=123465";

//khai báo đối tượng kết nối

SqlConnection con;

* *Sự kiện Load của frmEditEmployee*

//khởi tạo và mở connection

con = new SqlConnection(sqlconnect);

con.Open();

* *Sự kiện Click của nút Cập nhật*

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

{

try

{

//tạo đối tượng command

SqlCommand cmd = con.CreateCommand();

//thiết lập thủ thục update

cmd.CommandText = "UpdateEmployee";

//thiết lập loại câu lệnh

cmd.CommandType = CommandType.StoredProcedure;

//tạo tham số và gán giá trị cho tham số

cmd.Parameters.Add("@EmpId", SqlDbType.VarChar, 5).Value = txtId.Text;

cmd.Parameters.Add("@FirstName", SqlDbType.NVarChar, 20).Value = txtFirstName.Text;

cmd.Parameters.Add("@LastName", SqlDbType.NVarChar, 20).Value = txtLastName.Text;

cmd.Parameters.Add("@Sex", SqlDbType.Bit).Value = chkSex.Checked;

cmd.Parameters.Add("@Birthday", SqlDbType.DateTime).Value = txtBirthday.Value;

cmd.Parameters.Add("@Address", SqlDbType.NVarChar, 100).Value = txtAddress.Text;

cmd.Parameters.Add("@Email", SqlDbType.NVarChar, 100).Value = txtEmail.Text;

//thực thi câu truy vấn

cmd.ExecuteNonQuery();

MessageBox.Show("Cập nhật thành công!", "Cập nhật");

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message, "Insert", MessageBoxButtons.OK,

MessageBoxIcon.Information);

}

}

* *Sự kiện Click của nút Xóa*

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

{

//hỏi trước khi xóa

if (MessageBox.Show("Bạn có muốn xóa không?", "Delete data",

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

{

//tạo đối tượng

SqlCommand cmd = con.CreateCommand();

//tạo câu lệnh command

cmd.CommandText = "delete from Employee where EmpId=@EmpId";

//gán tham số

cmd.Parameters.Add("@EmpId", SqlDbType.VarChar, 5).Value = txtId.Text;

//kiểm tra xem xóa thành công không?

if (cmd.ExecuteNonQuery() != 1)

MessageBox.Show("Không tồn tại sinh viên có mã như trên", "Xóa",

MessageBoxButtons.OK, MessageBoxIcon.Error);

else

//xóa trắng form

txtId.Text = txtFirstName.Text = txtLastName.Text = txtEmail.Text = txtAddress.Text = "";

chkSex.Checked = false;

}

}

* *Sự kiện Leave của txtId*

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

{

//tạo đối tượng command

SqlCommand cmd = con.CreateCommand();

//thiết lập câu lệnh đọc nhân viên theo id

cmd.CommandText = "select \* from Employee where EmpId=@EmpId";

//gán giá trị cho tham số

cmd.Parameters.Add("@EmpId", SqlDbType.VarChar, 5).Value = txtId.Text;

//thực thi và trả về đối tượng reader

SqlDataReader dr = cmd.ExecuteReader();

//kiểm tra xem có dữ liệu không?

if (dr.HasRows)

{

//đọc dữ liệu ra form

if (dr.Read())

{

txtFirstName.Text = dr.GetString(1);

txtLastName.Text = dr.GetString(2);

chkSex.Checked = dr.GetBoolean(3);

txtBirthday.Value = dr.GetDateTime(4);

txtAddress.Text = dr.GetString(5);

txtEmail.Text = dr.GetString(6);

}

}

else

{

//thông báo khi không có dữ liệu

MessageBox.Show("Không tồn tại sinh viên có mã như trên", "Read data",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtFirstName.Text = txtLastName.Text = txtEmail.Text = txtAddress.Text = "";

chkSex.Checked = false;

}

//đóng đối tượng reader

dr.Close();

}

* *Sự kiện Click của nút Exit*

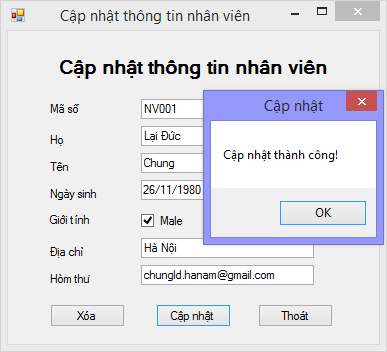
private void btnExit\_Click(object sender, EventArgs e)

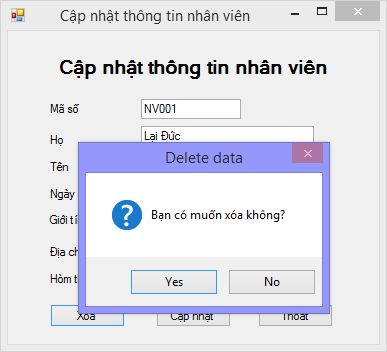
{

this.Close();

}

**Bước 4:** F5 để chạy và kiểm thử

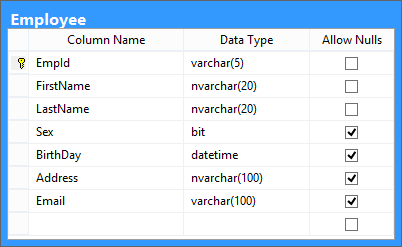




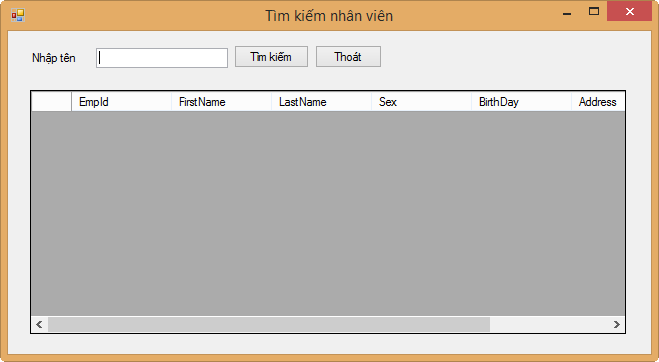
**Bài 4.3**

**Tạo ứng dụng Windows Form tìm kiếm nhân viên theo tên**

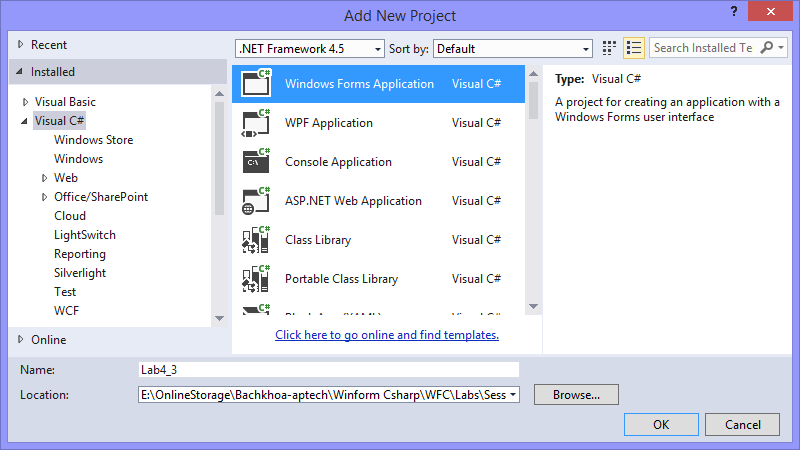
* *Cho cơ sở dữ liệu LabWinform với bảng cơ cấu trúc như sau*



* *Giao diện màn hình tìm kiếm nhân viên*



**Bước 1:** Kích chuột phải vào Solution “Session4” -> Add -> New Project.. -> đặt tên như hình



**Bước 2:** Thiết kế form theo đầu bài yêu cầu và đặt tên cho các control theo bảng sau (chú ý các control label không cần đặt tên):

|  |  |  |
| --- | --- | --- |
| **Loại Control** | **Đặt tên** | **Thuộc tính khác** |
| Màn hình: Form | frmSearch | Text:Tìm kiếm nhân viên |
| Mã số: TextBox | txtName |  |
| Kết quả: DataGridView | dgvEmployee |  |
| Tìm kiếm: Button | btnSearch | Text:Tìm kiếm |
| Thoát: Button | btnExit | Text: Thoát |

**Bước 3:** Code cho chương trình:

* *Đưa thư viện vào chương trình*

using System.Data.SqlClient;

* *Khai báo biến*

//khai báo chuỗi kết nối

string sqlconnect = "server=CHUNGLD\\SQLEXPRESS;database=LabWinform;uid=sa;pwd=123465";

//khai báo đối tượng DataSet

DataSet ds;

//Khai báo DataView

DataView dv;

* *Sự kiện Load của frmSearch*

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

{

//đọc dữ liệu ra dataset

SqlDataAdapter da = new SqlDataAdapter("select \* from employee", sqlconnect);

ds = new DataSet();

da.Fill(ds);

//đưa dữ liệu từ dataset vào dataview

dv = new DataView(ds.Tables[0]);

dgvEmployee.DataSource = dv;

}

* *Sự kiện Click của nút Tìm kiếm*

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

{

//Chỉ ra biểu thức lọc dữ liệu

dv.RowFilter = "LastName like '\*" + txtName.Text + "\*'";

dgvEmployee.DataSource = dv;

}

* *Sự kiện Click của nút Exit*

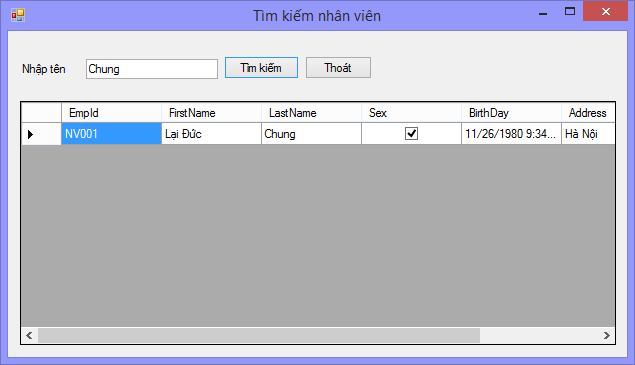
private void btnExit\_Click(object sender, EventArgs e)

{

this.Close();

}

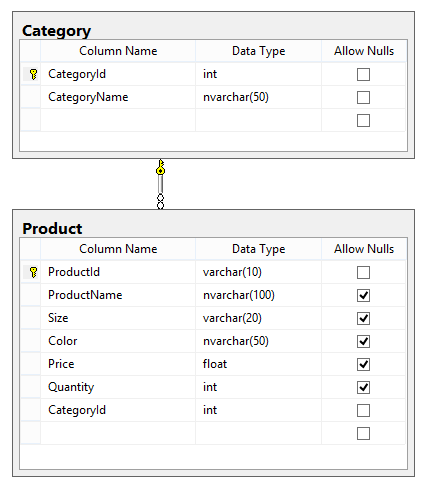
**Bước 4:** F5 để chạy và kiểm thử



**Bài 4.4**

**Tạo ứng dụng Windows Form cho phép hiển thị danh sách sản phẩm theo từng loại (dạng combobox và datagridview)**

* *Cho cơ sở dữ liệu LabWinform với bảng có cấu trúc như sau*



* *Dữ liệu mẫu*

insert into Category values(N'Quần áo')

insert into Category values(N'Nước hoa')

insert into Category values(N'Dầu gội')

insert into Category values(N'Xà bông')

insert into Product values('A001',N'Áo thu đông nữ','40-45',N'Trắng',200000,5,1)

insert into Product values('Q001',N'Quần jean france','28-30',N'Xanh',500000,10,1)

insert into Product values('A002',N'Áo cà xa','40-42',N'Đỏ',2000000,5,1)

insert into Product values('A003',N'Áo Nam','38-40',N'Xanh dương',220000,5,1)

insert into Product values('Q002',N'Quần Nam','28-30',N'Đen',280000,3,1)

insert into Product values('A004',N'Áo dài','40-42',N'Trắng',2100000,13,1)

insert into Product values('N001',N'Nước hoa nữ hoàng','200ml',N'Vàng',3500000,2,2)

insert into Product values('N002',N'Nước hoa pháp','120ml',N'Vàng',2400000,4,2)

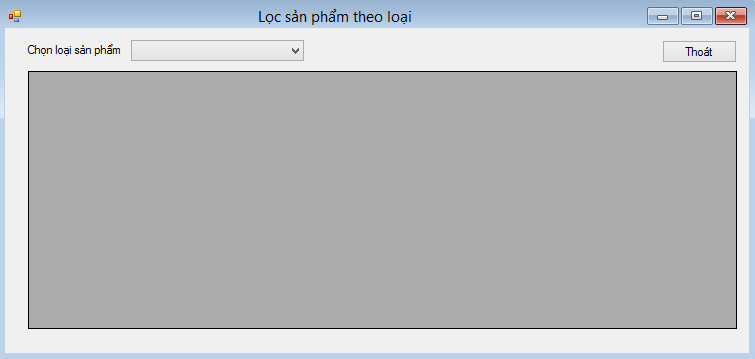
insert into Product values('N003',N'Nước hoa trung quốc','500ml',N'Vàng',300000,5,2)

insert into Product values('D001',N'Dầu gội thái lan','200ml',N'Trắng',75000,20,3)

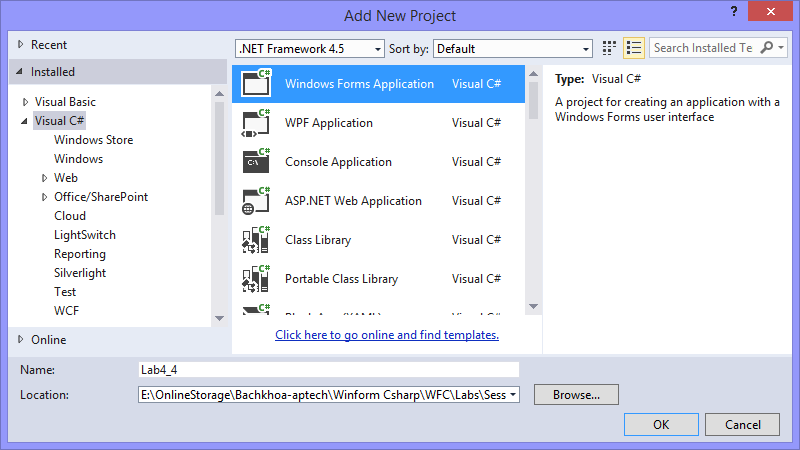
insert into Product values('D002',N'Dầu gội rejoin','500ml',N'Trắng',24000,4,3)

insert into Product values('D003',N'Dầu gội Clear','1000ml',N'Xanh',30000,5,3)

* *Giao diện màn lọc sản phẩm theo loại*



**Bước 1:** Kích chuột phải vào Solution “Session4” -> Add -> New Project.. -> đặt tên như hình



**Bước 2:** Thiết kế form theo đầu bài yêu cầu và đặt tên cho các control theo bảng sau (chú ý các control label không cần đặt tên):

|  |  |  |
| --- | --- | --- |
| **Loại Control** | **Đặt tên** | **Thuộc tính khác** |
| Màn hình: Form | frmProduct | Text:Lọc sản phẩm theo loại |
| Loại sản phẩm: ComboBox | cboCategory | DropDownStyle: DropDownList |
| Kết quả: DataGridView | dgvProduct |  |
| Thoát: Button | btnExit | Text: Thoát |

**Bước 3:** Code cho chương trình:

* *Đưa thư viện vào chương trình*

using System.Data.SqlClient;

* *Khai báo biến*

//khai báo chuỗi kết nối

string sqlconnect = "server=CHUNGLD\\SQLEXPRESS;database=LabWinform;uid=sa;pwd=123465";

* *Sự kiện Load của frmProduct*

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

{

try

{

//tạo đối tượng dataadapter

SqlDataAdapter da = new SqlDataAdapter("select \* from Category", sqlconnect);

//tạo dataset

DataSet ds = new DataSet();

//fill dữ liệu từ dataadapter ra dataset

da.Fill(ds, "Category");

//bind dữ liệu lên combo loại

cboCategory.DataSource = ds.Tables["Category"];

cboCategory.ValueMember = "CategoryId";

cboCategory.DisplayMember = "CategoryName";

//nếu có dữ liệu trong combo thì hiển thị sản phẩm lên lưới

if (cboCategory.Items.Count > 0)

DisplayProduct();

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message, "Display category", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

* *Sự kiện SelectionChangeCommitted của ComboBox*

private void DisplayProduct()

{

try

{

//tạo đối tượng dataadapter đọc dữ liệu từ bảng product theo category

SqlDataAdapter da = new SqlDataAdapter("select \* from Product where categoryid=@CategoryId", sqlconnect);

//gán tham số cho categoryid

da.SelectCommand.Parameters.Add("@categoryid",

SqlDbType.Int).Value = cboCategory.SelectedValue;

//tạo dataset

DataSet ds = new DataSet();

//fill dữ liệu vào dataset

da.Fill(ds, "Product");

//hiển thị lên lưới

dgvProduct.DataSource = ds.Tables["Product"];

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message, "Display product", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

* *Sự kiện Click của nút Exit*

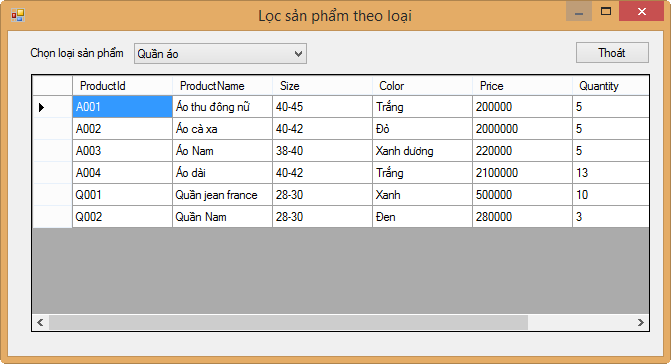
private void cboCategory\_SelectionChangeCommitted(object sender, EventArgs e)

{

DisplayProduct();

}

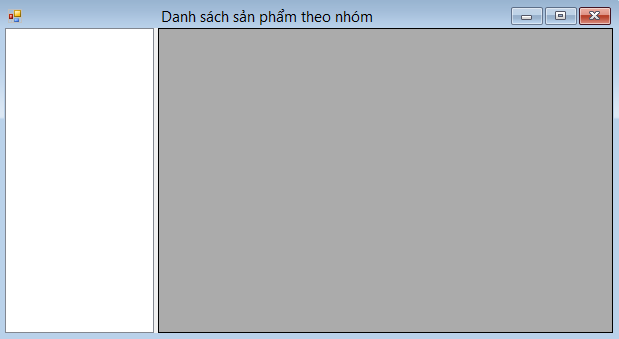
**Bước 4:** F5 để chạy và kiểm thử



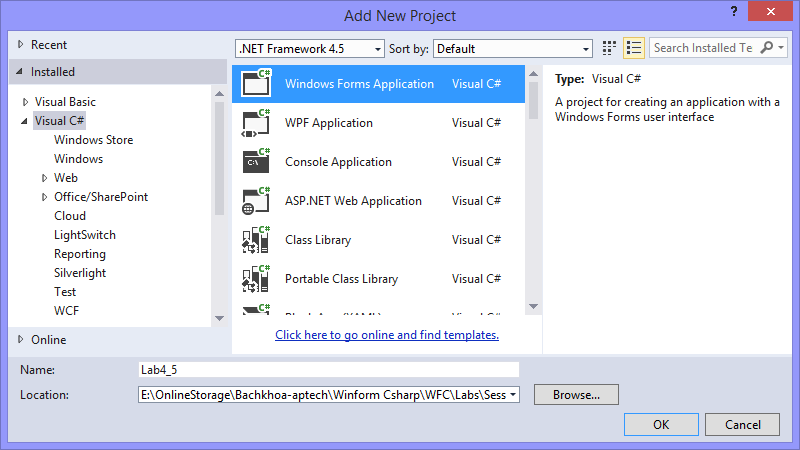
**Bài 4.5**

**Tạo ứng dụng Windows Form cho phép hiển thị sản phẩm theo loại (dạng treeview và datagridview)**

* *Cho cơ sở dữ liệu LabWinform giống bài 4.4*
* *Giao diện màn lọc sản phẩm theo loại*



**Bước 1:** Kích chuột phải vào Solution “Session4” -> Add -> New Project.. -> đặt tên như hình



**Bước 2:** Thiết kế form theo đầu bài yêu cầu và đặt tên cho các control theo bảng sau (chú ý các control label không cần đặt tên):

|  |  |  |
| --- | --- | --- |
| **Loại Control** | **Đặt tên** | **Thuộc tính khác** |
| Màn hình: Form | frmProduct | Text:Danh sách sản phẩm theo nhóm |
| ImageList | imgIcon |  |
| TreeView | trvCategory | ImageList:imgIcon |
| DataGridView | dgvProduct |  |

**Bước 3:** Code cho chương trình:

* *Đưa thư viện vào chương trình*

using System.Data.SqlClient;

* *Khai báo biến*

//khai báo chuỗi kết nối

string sqlconnect = "server=CHUNGLD\\SQLEXPRESS;database=LabWinform;uid=sa;pwd=123465";

//khai báo đối tượng kết nối

SqlConnection con;

* *Sự kiện Load của frmProduct*

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

{

//khởi tạo kết nối

con = new SqlConnection(sqlconnect);

con.Open();

//hiển thị danh mục

DisplayCategory();

//mở cây

trvCategory.ExpandAll();

}

* *Sự kiện AfterSelect của TreeView*

private void trvCategory\_AfterSelect(object sender, TreeViewEventArgs e)

{

//hiển thị sản phẩm khi kích vào nút trên cây

DisplayProduct();

}

//phương thức hiển thị danh mục

private void DisplayCategory()

{

try

{

//đọc dữ liệu ra datareader

SqlCommand cmd = con.CreateCommand();

cmd.CommandText = "select \* from category";

SqlDataReader dr = cmd.ExecuteReader();

//tạo nút gốc

TreeNode root = new TreeNode("Category", 0, 0);

root.Tag = "0";

//kiểm tra nếu có dữ liệu thì đọc ra nút

if (dr.HasRows)

{

while (dr.Read())

{

TreeNode node = new TreeNode(dr.GetString(1), 1, 1);

node.Tag = dr.GetValue(0);

//đưa nút đọc được vào nút gốc

root.Nodes.Add(node);

}

}

//đưa nút gốc lên cây

trvCategory.Nodes.Add(root);

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message, "Hiển thị danh mục",MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

//phương thức hiển thị sản phẩm

private void DisplayProduct()

{

try

{

//đọc dữ liệu theo category id

SqlDataAdapter da = new SqlDataAdapter("select \* from Product where " +

"categoryid=@CategoryId", sqlconnect);

da.SelectCommand.Parameters.Add("@categoryid",

SqlDbType.Int).Value = trvCategory.SelectedNode.Tag;

DataSet ds = new DataSet();

da.Fill(ds, "Product");

//hiển thị dữ liệu lên lưới

dgvProduct.DataSource = ds.Tables["Product"];

}

catch (SqlException ex)

{

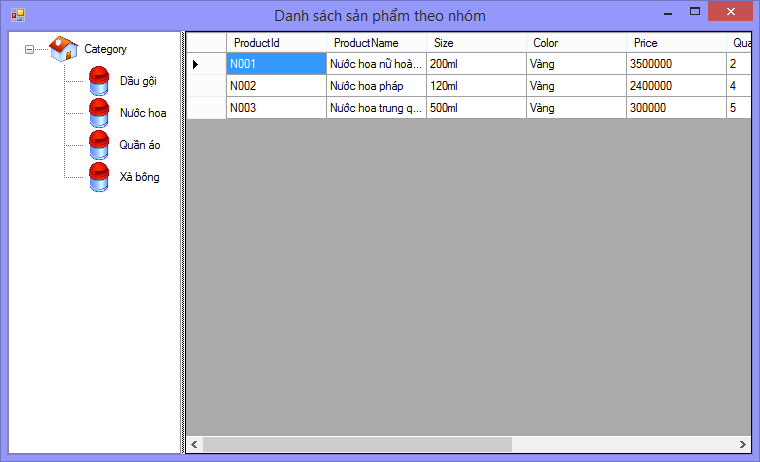
MessageBox.Show(ex.Message, "Hiển thị sản phẩm",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

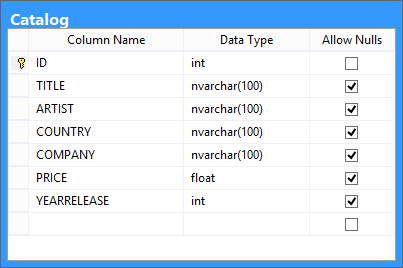
**Bước 4:** F5 để chạy và kiểm thử



**Bài 4.6**

**Tạo ứng dụng Windows Form cập nhật thông tin cho bảng catalog (sử dụng binding data)**

* *Cho cơ sở dữ liệu LabWinform với bảng có cấu trúc như sau*



* *Dữ liệu mẫu*

insert into Catalog values(N'Empire Burlesque',N'Bob Dylan',N'USA',N'Columbia',10.90,1985)

insert into Catalog values(N'Hide your heart',N'Bonnie Tyler',N'UK',N'CBS Records',9.90,1988)

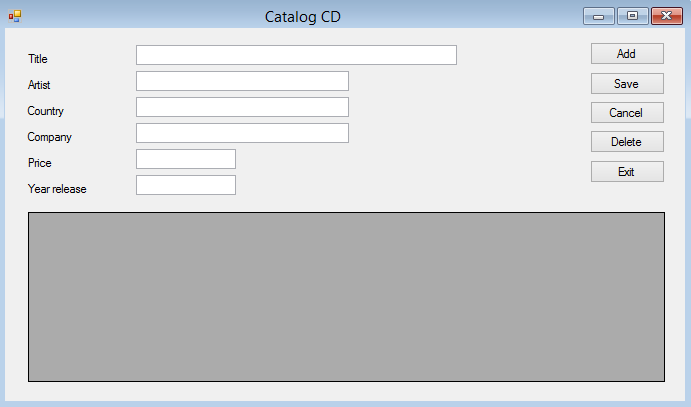
insert into Catalog values(N'Greatest Hits',N'Dolly Parton',N'USA',N'RCA',9.90,1982)

insert into Catalog values(N'Still got the blues',N'Gary Moore',N'UK',N'Virgin records',10.20,1990)

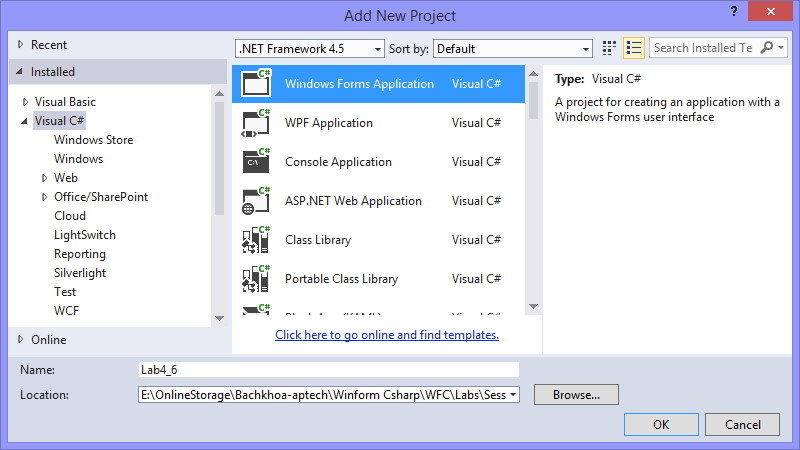
insert into Catalog values(N'Eros',N'Eros Ramazzotti',N'EU',N'BMG',9.90,1997)

insert into Catalog values(N'One night only',N'Bee Gees',N'UK',N'Polydor',10.90,1998)

* *Giao diện màn hình cập nhật thông tin catalog*



**Bước 1:** Kích chuột phải vào Solution “Session4” -> Add -> New Project.. -> đặt tên như hình



**Bước 2:** Thiết kế form theo đầu bài yêu cầu và đặt tên cho các control theo bảng sau (chú ý các control label không cần đặt tên):

|  |  |  |
| --- | --- | --- |
| **Loại Control** | **Đặt tên** | **Thuộc tính khác** |
| Màn hình: Form | frmCatalog | Text:Catalog CD |
| Title: TextBox | txtTitle |  |
| Artist: TextBox | txtArtist |  |
| Country: TextBox | txtCountry |  |
| Company:TextBox | txtCompany |  |
| Price: TextBox | txtPrice |  |
| Year Release: TextBox | txtRelease |  |
| Add: Button | btnAdd |  |
| Save: Button | btnSave |  |
| Cancel: Button | btnCancel |  |
| Delete: Button | btnDelete |  |
| Exit: Button | btnExit |  |
| DataGridView | dgvCatalog |  |

**Bước 3:** Code cho chương trình:

* *Đưa thư viện vào chương trình*

using System.Data.SqlClient;

* *Khai báo biến*

//khai báo chuỗi kết nối

string sqlconnect = "server=CHUNGLD\\SQLEXPRESS;database=LabWinform;uid=sa;pwd=123465";

//khai báo các biến khác

SqlDataAdapter da;

DataSet ds;

BindingManagerBase bmb;

* *Sự kiện Load của frmCatalog*

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

{

//gọi phương thức binding dữ liệu lên form

BindingCategory();

}

private void BindingCategory()

{

//lấy dữ liệu ra dataapdapter

da = new SqlDataAdapter("select \* from Catalog", sqlconnect);

//tạo đối tượng xây dựng các câu lệnh

new SqlCommandBuilder(da);

//tạo dataset

ds = new DataSet();

//fill dữ liệu ra dataset

da.Fill(ds);

//binding dữ liệu tới các control trên form

txtTitle.DataBindings.Add("Text", ds.Tables[0], "Title");

txtArtist.DataBindings.Add("Text", ds.Tables[0], "Artist");

txtCountry.DataBindings.Add("Text", ds.Tables[0], "Country");

txtCompany.DataBindings.Add("Text", ds.Tables[0], "Company");

txtPrice.DataBindings.Add("Text", ds.Tables[0], "Price");

txtRelease.DataBindings.Add("Text", ds.Tables[0], "YearRelease");

dgvCategory.DataSource = ds.Tables[0];

//đưa dữ liệu vào BindingContext để quản lý

bmb = BindingContext[ds.Tables[0]];

}

* *Sự kiện Click của nút Add*

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

{

//thêm mới

bmb.AddNew();

txtTitle.Focus();

}

* *Sự kiện Click của nút Save*

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

{

//lưu trữ

bmb.EndCurrentEdit();

da.Update(ds);

ds.AcceptChanges();

}

* *Sự kiện Click của nút Xóa*

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

{

//xóa

if (MessageBox.Show("Bạn có muốn xóa không?", "xóa bản ghi",

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

{

bmb.RemoveAt(bmb.Position);

da.Update(ds);

ds.AcceptChanges();

}

}

* *Sự kiện Click của nút Cancel*

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

{

//hủy bỏ

bmb.CancelCurrentEdit();

}

* *Sự kiện Click của nút Exit*

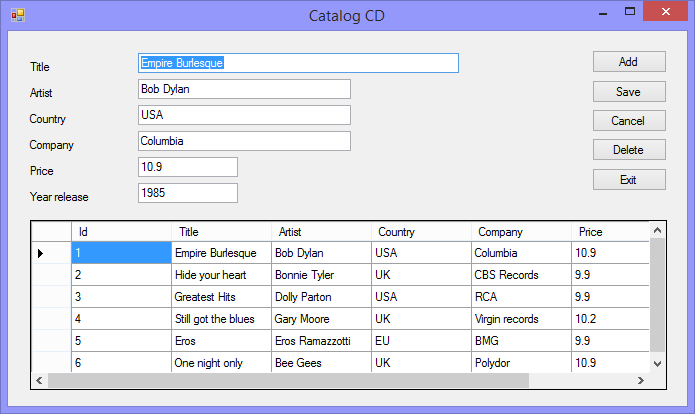
private void btnExit\_Click(object sender, EventArgs e)

{

this.Close();

}

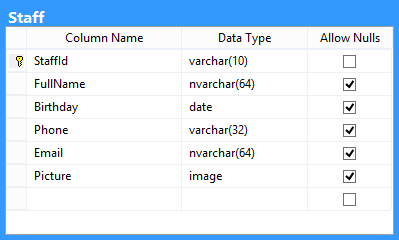
**Bước 4:** F5 để chạy và kiểm thử



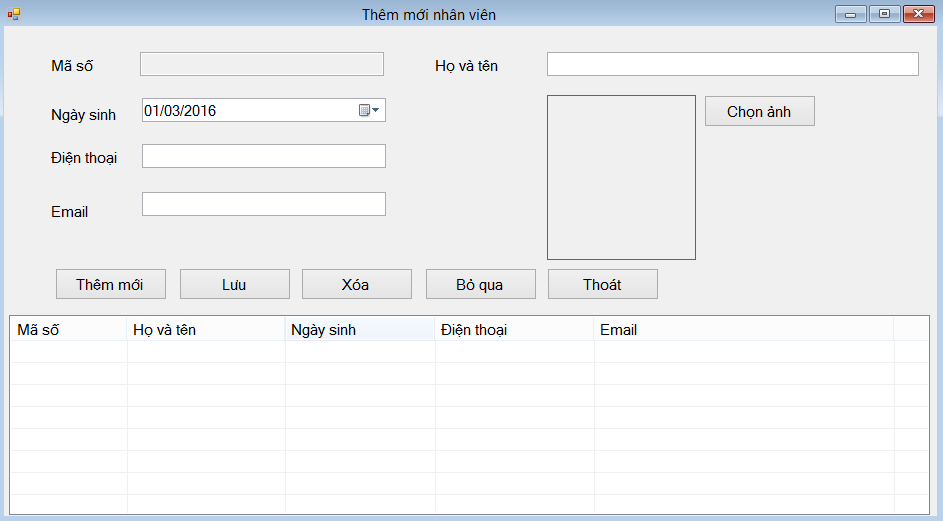
**Bài 4.7**

**Tạo ứng dụng Windows Form cập nhật thông tin cho bảng Staff (lưu ý trường Picture sẽ lưu hình ảnh vào bảng)**

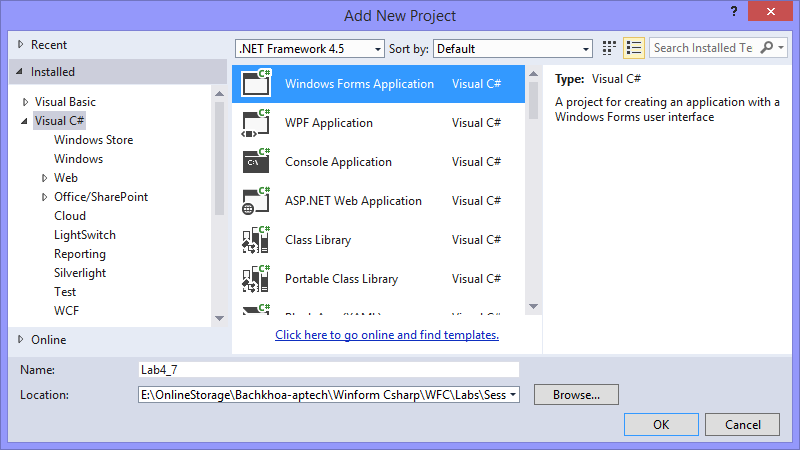
* *Cho cơ sở dữ liệu LabWinform với bảng có cấu trúc như sau*



* *Giao diện màn hình cập nhật thông tin Staff*



**Bước 1:** Kích chuột phải vào Solution “Session4” -> Add -> New Project.. -> đặt tên như hình



**Bước 2:** Thiết kế form theo đầu bài yêu cầu và đặt tên cho các control theo bảng sau (chú ý các control label không cần đặt tên):

|  |  |  |
| --- | --- | --- |
| **Loại Control** | **Đặt tên** | **Thuộc tính khác** |
| Màn hình: Form | frmStaff | Text:Cập nhật thông tin nhân viên |
| Mã số: TextBox | txtId |  |
| Họ và tên: TextBox | txtFullName |  |
| Ngày sinh: DateTimePicker | txtBirthday | CustomFormat: dd/MM/yyyy  Format: Custom |
| Điện thoại:TextBox | txtPhone |  |
| Email: TextBox | txtMail |  |
| Hình ảnh: PictureBox | ptPicture | SizeMode: zoom |
| Thêm mới: Button | btnAdd |  |
| Lưu: Button | btnSave |  |
| Bỏ qua: Button | btnCancel |  |
| Xóa: Button | btnDelete |  |
| Chọn ảnh: Button | btnSelectPicture |  |
| Thoát: Button | btnExit |  |
| ImageList | imgPicture | ColorDepth: 32bit  ImageSize: 100,120 |
| ListView | lstStaff | View: Details  Thiết kế các cột như hình |

**Bước 3:** Code cho chương trình:

* *Đưa thư viện vào chương trình*

using System.Data.SqlClient;

* *Khai báo biến*

//khai báo biến trạng thái sửa

bool edit = true;

//khai báo vị trí bản ghi

int position = -1;

//khai báo chuỗi kết nối

string strcon = "server=chungld\\sqlexpress;database = labwinform; uid=sa;pwd=123465";

//khai báo đối tượng kết nối

SqlConnection con;

* *Sự kiện frmStaff\_Load*

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

{

//tạo kết nối

con = new SqlConnection(strcon);

//nếu chưa mở thì mở

if (con.State != ConnectionState.Open)

con.Open();

//gọi phương thức đọc thông tin nhân viên

ReadStaff();

}

//phương thức đọc dánh sách staff

private void ReadStaff()

{

//Tạo command

SqlCommand cmd = con.CreateCommand();

//gán câu lệnh select

cmd.CommandText = "select \* from Staff";

//thực thi đọc dữ liệu

SqlDataReader dr = cmd.ExecuteReader();

//duyệt để đọc

while (dr.Read())

{

//tạo 1 listview item

ListViewItem item = new ListViewItem();

//gán các thông tin

item.Text = dr.GetString(0);

if (dr.GetValue(5) != DBNull.Value)

imgPicture.Images.Add(dr.GetString(0), Image.FromStream(dr.GetStream(5)));

item.SubItems.Add(dr.GetString(1));

item.SubItems.Add(dr.GetDateTime(2).ToString("dd/MM/yyyy"));

item.Tag = dr.GetDateTime(2);

item.SubItems.Add(dr.GetString(3));

item.SubItems.Add(dr.GetString(4));

//thêm vào listview

lstStaff.Items.Add(item);

}

//đóng bộ đọc

dr.Close();

//nếu có dữ liệu thì hiển thị dòng đầu tiên

if (lstStaff.Items.Count > 0)

{

position = 0;

lstStaff.Items[position].Selected = true;

lstStaff.Select();

ShowDetailStaff();

}

}

* *Sự kiện click cho nút thêm mới*

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

{

//Xóa các thông tin trên form

txtId.Text = txtFullName.Text = txtMail.Text = txtPhone.Text = "";

ptPicture.Image = null;

//đặt trạng thái sửa là false

edit = false;

//cho phép nhập dữ liệu vào ô id

txtId.ReadOnly = false;

//đặt con trỏ vào ô Id

txtId.Focus();

}

* *Sự kiện click cho nút ghi*

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

{

try

{

//Tạo đối tượng Command

SqlCommand cmd = con.CreateCommand();

//nếu không phải sửa (tức là thêm mới)

if (!edit)

{

//gán chuỗi sql insert

cmd.CommandText = "insert into Staff values(@id, @name, @birthday,@phone,@email,@picture)";

}

else

{

//gán chuỗi sql update

cmd.CommandText = "update Staff set fullname= @name,birthday= @birthday,phone=@phone,email=@email,picture=@picture where staffid=@id";

}

//gán giá trị cho các tham số

cmd.Parameters.Add("@id", SqlDbType.VarChar, 10).Value = txtId.Text;

cmd.Parameters.Add("@name", SqlDbType.NVarChar, 64).Value = txtFullName.Text;

cmd.Parameters.Add("@birthday", SqlDbType.Date).Value = txtBirthday.Value;

cmd.Parameters.Add("@phone", SqlDbType.VarChar, 32).Value = txtPhone.Text;

cmd.Parameters.Add("@email", SqlDbType.NVarChar, 64).Value = txtMail.Text;

//xử lý gán giá trị cho tham số ảnh

if (ptPicture.Image != null)

cmd.Parameters.Add("@picture", SqlDbType.Image).Value = new ConvertImage().ToByteArray(ptPicture.Image, ImageFormat.Jpeg);

else

cmd.Parameters.Add("@picture", SqlDbType.Image).Value = DBNull.Value;

//thực thi câu lệnh

cmd.ExecuteNonQuery();

//tạo phần tử mới

ListViewItem item = new ListViewItem();

item.Text = txtId.Text;

item.SubItems.Add(txtFullName.Text);

item.SubItems.Add(txtBirthday.Text);

item.Tag = txtBirthday.Value;

item.SubItems.Add(txtPhone.Text);

item.SubItems.Add(txtMail.Text);

//nếu có ảnh

if (ptPicture.Image != null)

{

//xóa ảnh cũ

imgPicture.Images.RemoveByKey(txtId.Text);

//thêm ảnh mới

imgPicture.Images.Add(txtId.Text, ptPicture.Image);

}

if (!edit) //nếu là thêm

{

//bỏ chọn tại vị trí trước

if (position >= 0)

lstStaff.Items[position].Selected = false;

//thêm item

lstStaff.Items.Add(item);

//chọn phần tử mới thêm ở cuối

lstStaff.Items[lstStaff.Items.Count - 1].Selected = true;

position = lstStaff.Items.Count - 1;

}

else

{

//xóa phần tử được chọn trong listview đi

if (position >= 0)

{

lstStaff.Items.Remove(lstStaff.Items[position]);

//chèn vào đúng vị trí sửa

lstStaff.Items.Insert(position, item);

//chọn phần tử vừa sửa

lstStaff.Items[position].Selected = true;

}

}

lstStaff.Select();

//không cho phép nhập dữ liệu vào ô id

txtId.ReadOnly = true;

edit = true;

}catch(Exception ex)

{

MessageBox.Show(ex.Message, "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

* *Sự kiện click cho nút xóa*

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

{

if (position >= 0)

{

//Hỏi trước khi xóa

if (MessageBox.Show("Bạn có muốn xóa không?", "Thông báo", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

//Tạo đối tượng command

SqlCommand cmd = con.CreateCommand();

//gán câu lệnh sql xóa

cmd.CommandText = "delete from Staff where StaffId=@id";

//truyền giá trị cho tham số

cmd.Parameters.Add("@id", SqlDbType.VarChar, 10).Value = txtId.Text;

//thực thi

cmd.ExecuteNonQuery();

//xóa hình ảnh

imgPicture.Images.RemoveByKey(txtId.Text);

//xóa item

lstStaff.Items.Remove(lstStaff.Items[position]);

//thiết lập lại dòng chọn

if (lstStaff.Items.Count > 0)

{

if (position == lstStaff.Items.Count)

position--;

lstStaff.Items[position].Selected = true;

lstStaff.Select();

ShowDetailStaff();

}

else

{

position = -1;

//Xóa các thông tin trên form

txtId.Text = txtFullName.Text = txtMail.Text = txtPhone.Text = "";

ptPicture.Image = null;

}

}

}

else

MessageBox.Show("Hãy chọn 1 dòng cần xóa", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

* *Sự kiện click cho ListView*

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

{

position = lstStaff.FocusedItem.Index;

//kích vào listview thì hiển thị chi tiết

ShowDetailStaff();

}

//phương thức hiển thị chi tiết nhân viên

private void ShowDetailStaff()

{

if (position >= 0)

{

//lấy item được chọn

ListViewItem item = lstStaff.Items[position];

//nếu có thì hiển thị chi tiết lên form

if (item != null)

{

txtId.Text = item.Text;

ptPicture.Image = imgPicture.Images[item.Text];

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

txtBirthday.Text = item.Tag.ToString();

txtMail.Text = item.SubItems[4].Text;

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

}

}

//không cho phép nhập dữ liệu vào ô id

txtId.ReadOnly = true;

edit = true;

}

* *Sự kiện click cho nút Bỏ qua*

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

{

//hiển thị chi tiết lên form

ShowDetailStaff();

lstStaff.Select();

}

* *Sự kiện click cho nút Chọn ảnh*

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

{

//tạo cửa sổ chọn tệp tin

OpenFileDialog file = new OpenFileDialog();

//lọc đuôi cần chọn

file.Filter = "(\*.JPG)|\*.JPG";

//hiển thị cửa sổ chọn

file.ShowDialog();

//nếu có tệp được chọn thì hiển thị ảnh lên control picturebox

if (file.FileName != "")

ptPicture.Image = Image.FromFile(file.FileName);

}

**Bước 4: Tạo lớp ConvertImage theo code gợi ý sau:**

class ConvertImage

{

public byte[] ToByteArray(System.Drawing.Image imageToConvert, System.Drawing.Imaging.ImageFormat formatOfImage)

{

byte[] Ret=null;

try

{

using (MemoryStream ms = new MemoryStream())

{

imageToConvert.Save(ms, formatOfImage);

Ret = ms.ToArray();

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "Thông báo lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

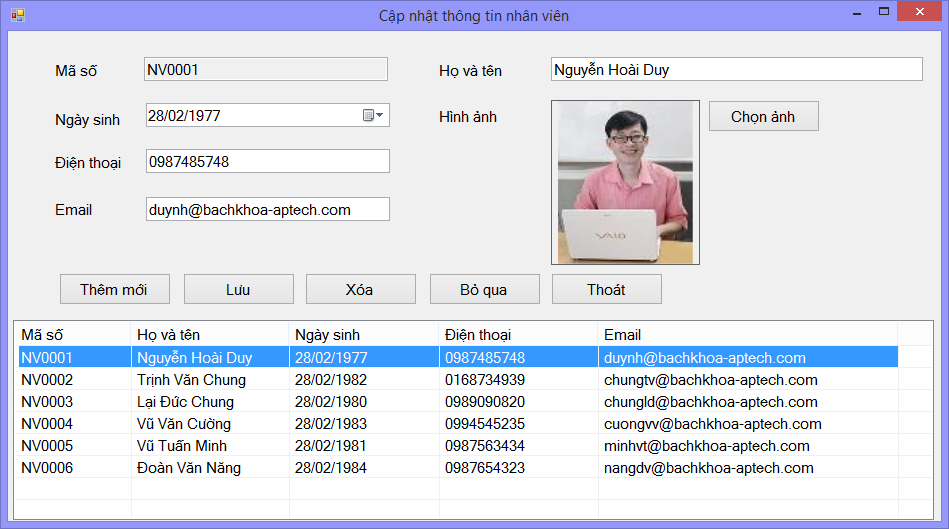
}

return Ret;

}

}

**Bước 5:** F5 để chạy và kiểm tra kết quả



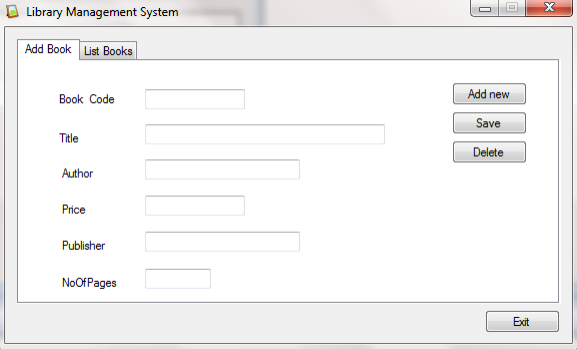
**Phần I: Bài tập tự làm**

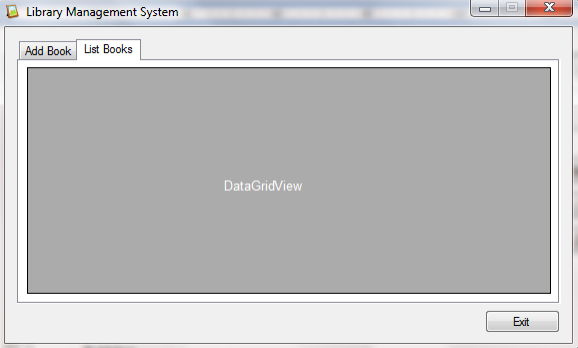
**Bài 4.1**:

* Cho cơ sở dữ liệu mẫu như sau:

-----------Tạo cơ sở dữ liệu LabWinform---------  
Create database LabWinform  
go  
use LabWinform  
go  
--------------Tạo bảng Book -------------------  
Create table Books  
(  
BookCode varchar(10) primary key,  
Title nvarchar(100),  
Author nvarchar(100),  
Price float,  
Publisher nvarchar(100),  
NoOfPages int  
)  
--Thêm một vài bản ghi

* Thiết kế màn hình nhập liệu và hiển thị dữ liệu theo mô tả sau:
  + Sử dụng Tab control để bố cục phân màn hình làm 2 trang
  + Trang 1 Add Book để thêm thông tin chi tiết về sách
  + Trang 2 List Boox sử dụng control DataGridView để hiển thị danh sách Book
  + Tại List Book, khi kích đúp vào quyển sách bất kỳ thì màn hình quay về trang Add Book hiển thị chi tiết để sửa.





**HẾT**