

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

{

try

{

SqlConnection conn = new SqlConnection(Program.strConn);

conn.Open();

String sSQL = "SELECT \* FROM Department";

SqlCommand cmd = new SqlCommand(sSQL, conn);

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

comboBoxDepartment cũng là 1 datasource. Chúng ta lấy dữ liệu từ trong data table, gắn vào trong data source của comboBoxDepartment

DisplayMemeber là những thành phần được hiển thị trong comboBoxDepartment

ValueMember là giá trị của những thành phần đó, **ValueMember sẽ là thứ được dùng để so sánh và láy giá trị**

DataTable dt = new DataTable();

adapter.Fill(dt);

if (dt.Rows.Count > 0)

{

comboBoxDepartment.DataSource = dt;

comboBoxDepartment.DisplayMember = "DName";

comboBoxDepartment.ValueMember = "DNumber";

LoadData();

}

else

{

MessageBox.Show("No Data", "Warnning");

}

adapter.Dispose();

}

catch (Exception ex)

{

//MessageBox.Show("Error: " + ex.Message + " - " + ex.Source);

}

}

private void LoadData()

{

try

{

SqlConnection conn = new SqlConnection(Program.strConn);

conn.Open();

String sSQL = "SELECT \* FROM Employee WHERE DNo=@CB";

SqlCommand cmd = new SqlCommand(sSQL, conn);

Chúng ta nói ValueMember sẽ là thứ được dùng để so sánh và lấy giá trị. Điều đó được áp dụng ở đây khi DNo=@CB, với @CB là comboBoxDepartment.SelectedValue, SelectedValue đây không phải là Name của Department mà chúng là mã số của Department

cmd.Parameters.AddWithValue("@CB", comboBoxDepartment.SelectedValue.ToString());

//String sSQL = "SELECT \* FROM Employee where DNo=" + comboBoxDepartment.SelectedValue;

//SqlCommand cmd = new SqlCommand(sSQL, conn);

//SqlCommand cmd = new SqlCommand("SELECT \* FROM Employee where DNo=" + comboBoxDepartment.SelectedValue, conn);

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

adapter.Fill(dt);

if (dt.Rows.Count > 0)

{

dataGridView1.DataSource = dt;

}

else

{

MessageBox.Show("No Data", "Warning");

}

}

catch (Exception ex)

{

//MessageBox.Show("Error: " + ex.Message + " - " + ex.Source);

}

}

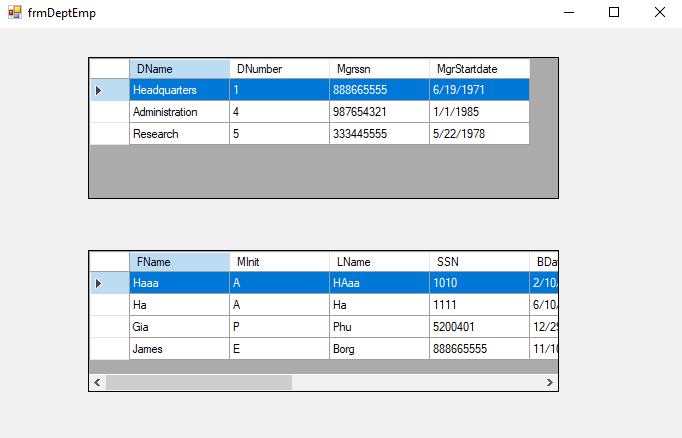
private void comboBoxDepartment\_SelectedIndexChanged(object sender, EventArgs e)

{

Data sẽ được load khi chúng ta thay đổi index trong comboBoxDepartment

LoadData();

}



private void LoadData()

{

try

{

//Get row index of a selected cell

//seletectedCell[0] là khóa luôn cái column, chỉ để tự do cái row, giá trị trả về là column và row

int index = dataGridViewDepartment.SelectedCells[0].RowIndex;

Index sẽ lấy giá trị của SelectedCell, chúng ta cố định giá trị column là 0 và chúng ta lấy RowIndex

if (index < 0 || index >= dataGridViewDepartment.RowCount)

{

MessageBox.Show("Please select a department first", "Warning");

return;

}

//MessageBox.Show("Test: " + index);

Lấy giá trị của nguyên 1 Row dựa vào index

Sau đó lấy giá trị của Cells

DataGridViewRow row = dataGridViewDepartment.Rows[index];

//Get DNumber from the dataGrid

int iDNo = int.Parse(row.Cells[1].Value.ToString());

SqlConnection conn = new SqlConnection(Program.strConn);

conn.Open();

String sSQL = "SELECT \* FROM Employee WHERE DNo=@CB";

SqlCommand cmd = new SqlCommand(sSQL, conn);

Câu lệnh SQL này có điều kiện WHERE => nó quan trọng và nó sẽ liên kết các bản với nhau

cmd.Parameters.AddWithValue("@CB", iDNo);

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

adapter.Fill(dt);

if(dt.Rows.Count > 0)

{

dataGridViewEmployee.DataSource = dt;

}

else

{

MessageBox.Show("No Data", "Warning");

}

adapter.Dispose();

dataGridViewEmployee.Refresh();

}

catch(Exception ex)

{

}

}

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

{

SqlConnection conn = new SqlConnection(Program.strConn);

conn.Open();

String sSQL = "SELECT \* FROM Department";

SqlCommand cmd = new SqlCommand(sSQL, conn);

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

Chúng ta để ý, những chổ nào lấy giá trị ra (dùng câu lệnh SELECT) thì có adapter. Những chỗ nào dùng những truy vấn/thay đổi thì không dùng

DataTable dt = new DataTable();

adapter.Fill(dt);

if(dt.Rows.Count > 0 )

{

dataGridViewDepartment.DataSource = dt;

}

else

{

MessageBox.Show("No Data", "Warning");

}

adapter.Dispose();

LoadData();

}

private void dataGridViewDepartment\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

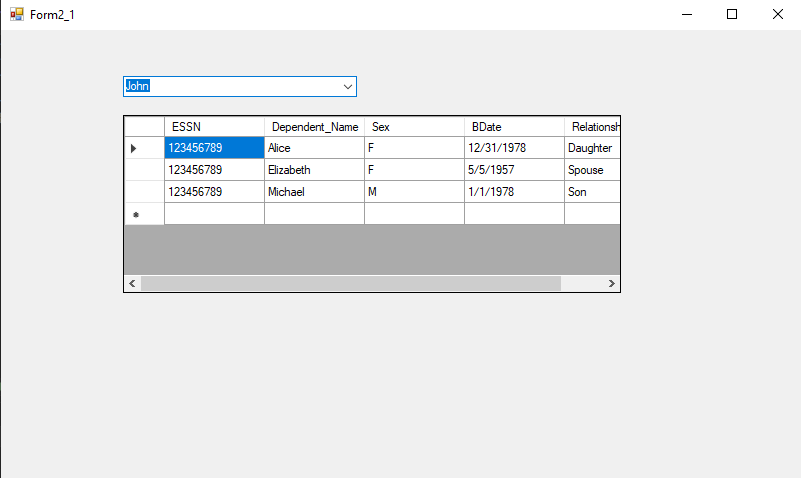
LoadData();

dataGridViewDepartment.Refresh();

dataGridViewEmployee.Refresh();

}

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------**



Bài này tương tự bài 1 nhưng chúng ta không dùng DNo để so sánh nữa mà chúng ta dùng SSN để so sánh và lấy giá trị

private void Form2\_1\_Load(object sender, EventArgs e)

{

try

{

SqlConnection conn = new SqlConnection(Program.strConn);

conn.Open();

String sSQL = "SELECT \* FROM Employee";

SqlCommand cmd = new SqlCommand(sSQL, conn);

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

Data source thì là at (bình thường

Display Name giờ là Fname

**Value Member (dùng để so sánh và lấy giá trị) giờ là SSN**

DataTable dt = new DataTable();

adapter.Fill(dt);

if (dt.Rows.Count > 0)

{

comboBoxEmployee.DataSource = dt;

comboBoxEmployee.DisplayMember = "FName";

comboBoxEmployee.ValueMember = "SSN";

LoadData();

}

else

{

MessageBox.Show("No Data 1", "Warnning");

}

adapter.Dispose();

}

catch (Exception ex)

{

//MessageBox.Show("Error: " + ex.Message + " - " + ex.Source);

}

}

private void LoadData()

{

try

{

SqlConnection conn = new SqlConnection(Program.strConn);

conn.Open();

String sSQL = "SELECT \* FROM Dependent WHERE ESSN=@CB";

SqlCommand cmd = new SqlCommand(sSQL, conn);

cmd.Parameters.AddWithValue("@CB", comboBoxEmployee.SelectedValue.ToString());

//String sSQL = "SELECT \* FROM Employee where DNo=" + comboBoxDepartment.SelectedValue;

//SqlCommand cmd = new SqlCommand(sSQL, conn);

//SqlCommand cmd = new SqlCommand("SELECT \* FROM Employee where DNo=" + comboBoxDepartment.SelectedValue, conn);

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

adapter.Fill(dt);

if (dt.Rows.Count > 0)

{

dataGridView1.DataSource = dt;

}

else

{

//MessageBox.Show("No Data", "Warning");

}

}

catch (Exception ex)

{

//MessageBox.Show("Error: " + ex.Message + " - " + ex.Source);

}

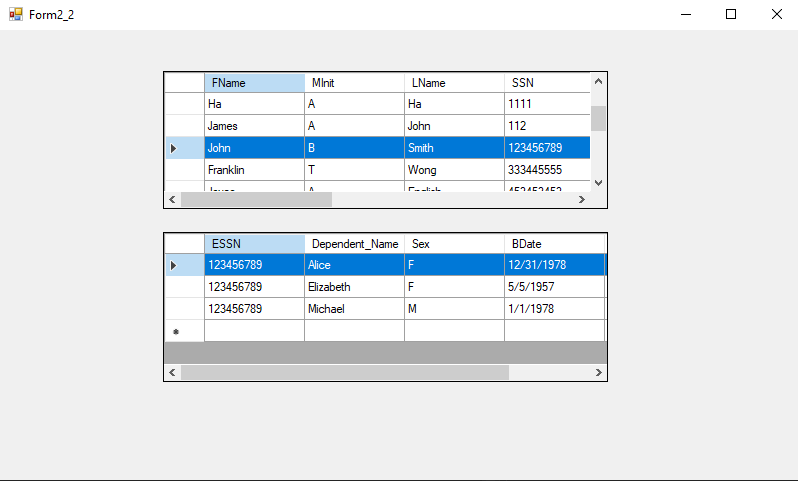
}

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

{

LoadData();

}



Bài này cũng tương tự bài 1 chỉ khác chúng ta lấy giá trị từ SSN và ESSN

private void LoadData()

{

try

{

//Get row index of a selected cell

//seletectedCell[0] là khóa luôn cái column, chỉ để tự do cái row, giá trị trả về là column và row

int index = dataGridView1.SelectedCells[0].RowIndex;

if (index < 0 || index >= dataGridView1.RowCount)

{

MessageBox.Show("Please select a department first", "Warning");

return;

}

//MessageBox.Show("Test: " + index);

DataGridViewRow row = dataGridView1.Rows[index];

//Get SSN from the dataGrid

int iDNo = int.Parse(row.Cells[3].Value.ToString());

SqlConnection conn = new SqlConnection(Program.strConn);

conn.Open();

String sSQL = "SELECT \* FROM Dependent WHERE ESSN=@CB";

SqlCommand cmd = new SqlCommand(sSQL, conn);

cmd.Parameters.AddWithValue("@CB", iDNo);

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

adapter.Fill(dt);

if (dt.Rows.Count > 0)

{

dataGridView2.DataSource = dt;

}

else

{

MessageBox.Show("No Data", "Warning");

}

adapter.Dispose();

dataGridView2.Refresh();

}

catch (Exception ex)

{

}

}

private void Form2\_2\_Load(object sender, EventArgs e)

{

SqlConnection conn = new SqlConnection(Program.strConn);

conn.Open();

String sSQL = "SELECT \* FROM Employee";

SqlCommand cmd = new SqlCommand(sSQL, conn);

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

adapter.Fill(dt);

if (dt.Rows.Count > 0)

{

dataGridView1.DataSource = dt;

}

else

{

MessageBox.Show("No Data", "Warning");

}

adapter.Dispose();

LoadData();

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

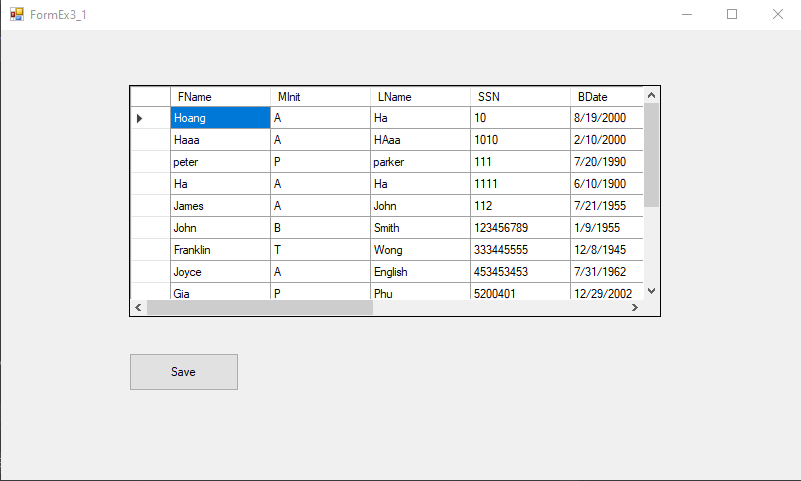
LoadData();

dataGridView1.Refresh();

dataGridView2.Refresh();

}

**------------------------------------------------------------------------------------------------------------------------------------------------------------------------**



Chúng ta để ý dòng khai báo global variable int db\_data\_rows\_count = -1

Dòng này được dùng để đến bên trong database trước khi add có bao nhiêu dòng dữ liệu

Bài này dùng index để lưu giá trị vào

Khác với solution 2 là dùng OleDB

OleDB nhìn dễ xử lý hơn nhưng cái code quen tay hơn

public partial class FormEx3\_1 : Form

{

int db\_data\_rows\_count = -1;

public FormEx3\_1()

{

InitializeComponent();

}

private void FormEx3\_1\_Load(object sender, EventArgs e)

{

LoadData();

}

private void LoadData()

{

try

{

SqlConnection conn = new SqlConnection(Program.strConn);

conn.Open();

String sSQL = "SELECT \* FROM Employee";

SqlCommand cmd = new SqlCommand(sSQL, conn);

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

adapter.Fill(dt);

if(dt.Rows.Count > 0)

{

dataGridView1.DataSource = dt;

}

else

{

MessageBox.Show("No Data", "Warning");

}

//Count how many rows are in database before everything else

db\_data\_rows\_count = dt.Rows.Count;

Sau khi database đã được đọc thì chúng ta sẽ gắn số lượng dòng vào trong đây

adapter.Dispose();

}

catch (Exception ex)

{

}

}

Vòng lặp sẽ chạy để add tất cả những dòng mới Edit vào, gặp dòng nào không có => dừng không thêm tiếp

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

{

try

{

for (int row\_index = db\_data\_rows\_count; row\_index < dataGridView1.Rows.Count; row\_index++)

{

if (dataGridView1.Rows[row\_index].Cells[0].Value == null)

{

break;

}

String FName = dataGridView1.Rows[row\_index].Cells[0].Value.ToString().Trim();

String Mint = (dataGridView1.Rows[row\_index].Cells[1].Value.ToString().Trim().Equals("")) ? "NULL" : dataGridView1.Rows[row\_index].Cells[1].Value.ToString().Trim();

String LName = dataGridView1.Rows[row\_index].Cells[2].Value.ToString().Trim();

String SSN = dataGridView1.Rows[row\_index].Cells[3].Value.ToString().Trim();

String BDate = (dataGridView1.Rows[row\_index].Cells[4].Value.ToString().Trim().Equals("")) ? "NULL" : dataGridView1.Rows[row\_index].Cells[4].Value.ToString().Trim();

String Address = (dataGridView1.Rows[row\_index].Cells[5].Value.ToString().Trim().Equals("")) ? "NULL" : dataGridView1.Rows[row\_index].Cells[5].Value.ToString().Trim();

String Sex = dataGridView1.Rows[row\_index].Cells[6].Value.ToString().Trim();

int Salary = int.Parse(dataGridView1.Rows[row\_index].Cells[7].Value.ToString().Trim());

String SuperSSN = dataGridView1.Rows[row\_index].Cells[8].Value.ToString().Trim();

int DNo = int.Parse(dataGridView1.Rows[row\_index].Cells[9].Value.ToString().Trim());

SqlConnection conn = new SqlConnection(Program.strConn);

conn.Open();

String sSQL = "INSERT INTO Employee VALUES (@FName, @Mint, @LName, @SSN, @BDate, @Address, @Sex, @Salary, @SuperSSN, @DNo)";

SqlCommand cmd = new SqlCommand(sSQL, conn);

cmd.Parameters.AddWithValue("@Fname", FName);

cmd.Parameters.AddWithValue("@Mint", Mint);

cmd.Parameters.AddWithValue("@LName", LName);

cmd.Parameters.AddWithValue("@SSN", SSN);

cmd.Parameters.AddWithValue("@BDate", BDate);

cmd.Parameters.AddWithValue("@Address", Address);

cmd.Parameters.AddWithValue("@Sex", Sex);

cmd.Parameters.AddWithValue("@Salary", Salary);

cmd.Parameters.AddWithValue("@SuperSSN", SuperSSN);

cmd.Parameters.AddWithValue("@DNo", DNo);

Chỗ này phải add cho cẩn thận, đúng số, vì SQL rất là “khó”

Ngoài ra, chúng ta cần chú ý đến ràng buộc như PK, FK,… .SQL sẽ tự động báo nếu chúng ta vi phạm những ràng buộc này trong quá trình thêm vào nhưng vẫn nên chú ý

int i = cmd.ExecuteNonQuery();

if (i != 0)

{

MessageBox.Show("Saved");

}

else

{

MessageBox.Show("Error");

}

}

}

catch (Exception ex)

{

//Database có primary key và foreign key, khi có lỗi xày ra sẽ báo lỗi và không cho add vào trong database

MessageBox.Show(ex.Message);

}

LoadData();

}

}