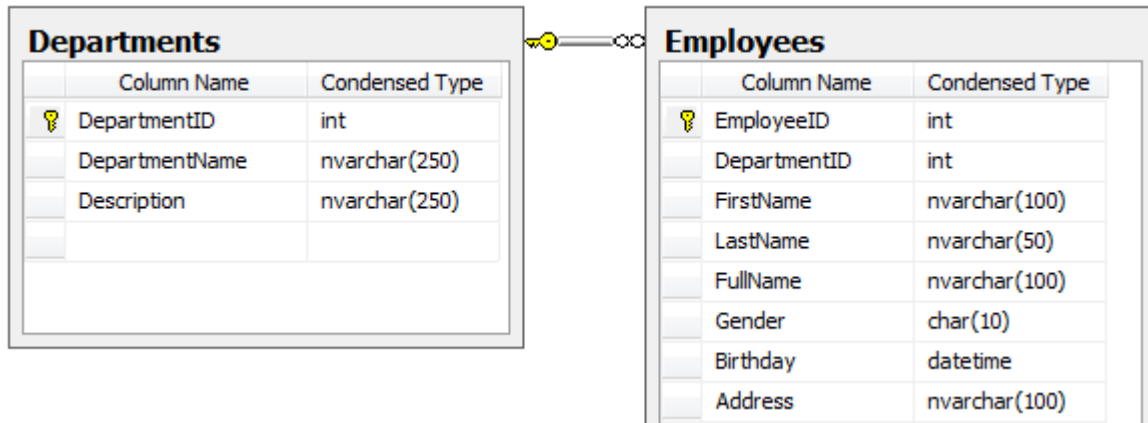
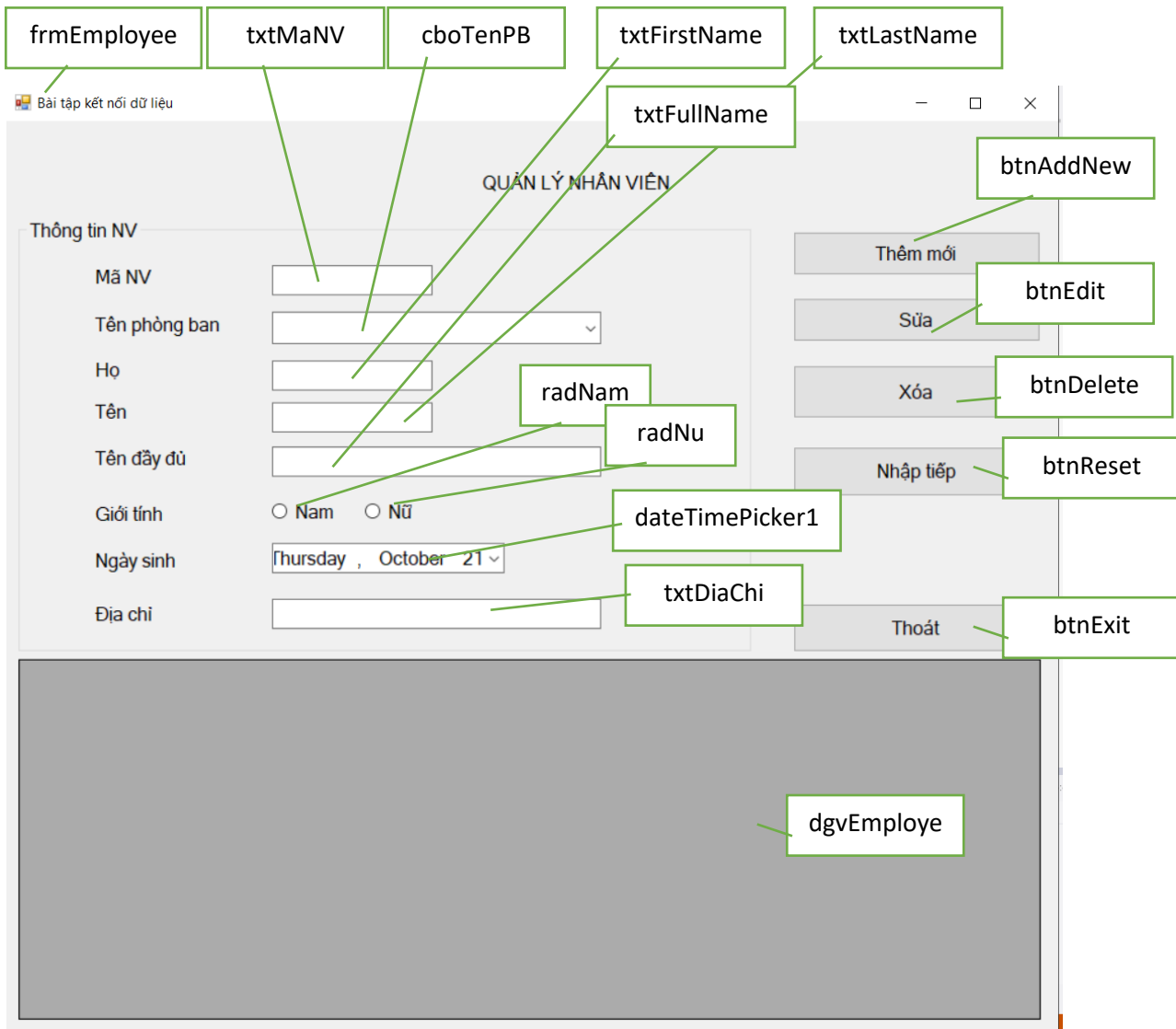


## BÀI TẬP THỰC HÀNH ADO

Bước 1. Mở SQL Server tạo 1 CSDL có tên QLNS với 2 bảng CSDL: Departments (Đơn vị) và Employees (Nhân viên). Có cấu trúc như sau:



Bước 2. Mở winform tạo 1 project mới, Thiết kế giao diện như sau:




### Bước 3. Lấy đường dẫn kết nối (connectionString)

- Trên Visual Studio -> Project -> Add New Data Source


Data Source Configuration Wizard


?


×

 **Choose a Data Source Type**

**Where will the application get data from?**

 Database

 Service

 Object

Lets you connect to a database and choose the database objects for your application.

< Previous

Next >


Finish

Cancel


Data Source Configuration Wizard


?

×

 **Choose a Database Model**

**What type of database model do you want to use?**

 Dataset

 Entity Data Model

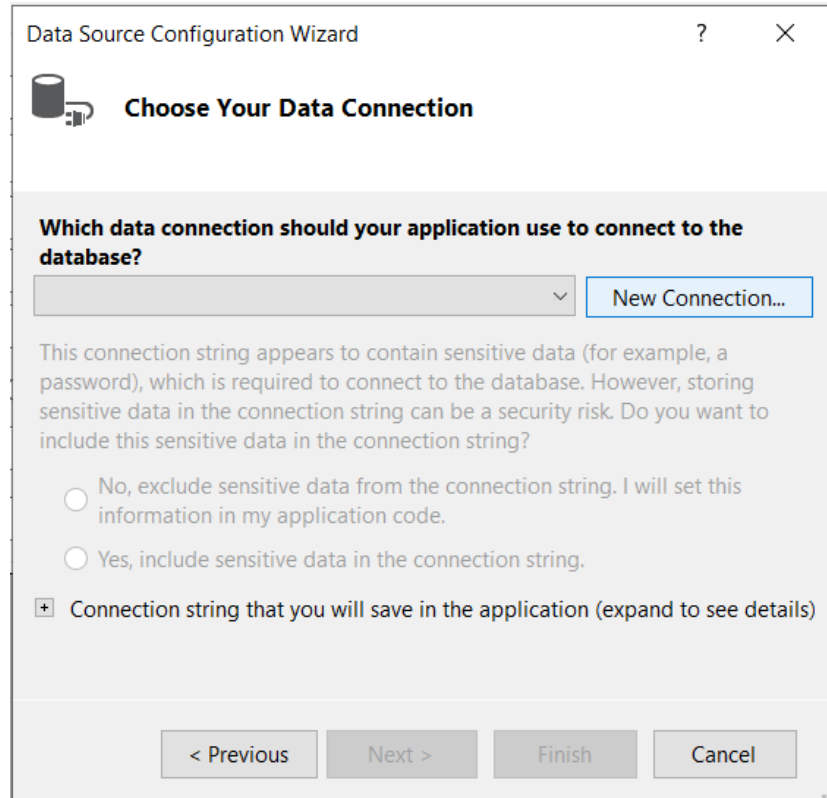
The database model you choose determines the types of data objects your application code uses. A dataset file will be added to your project.

< Previous

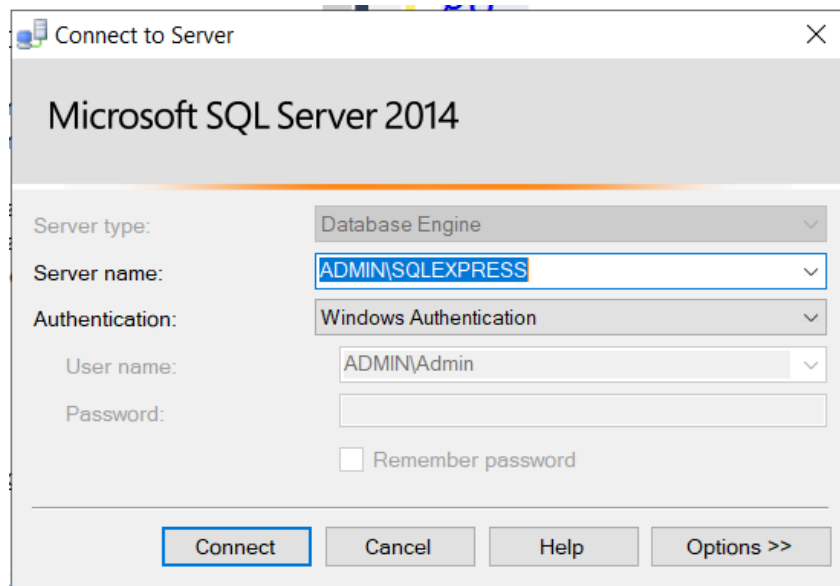
Next >

Finish

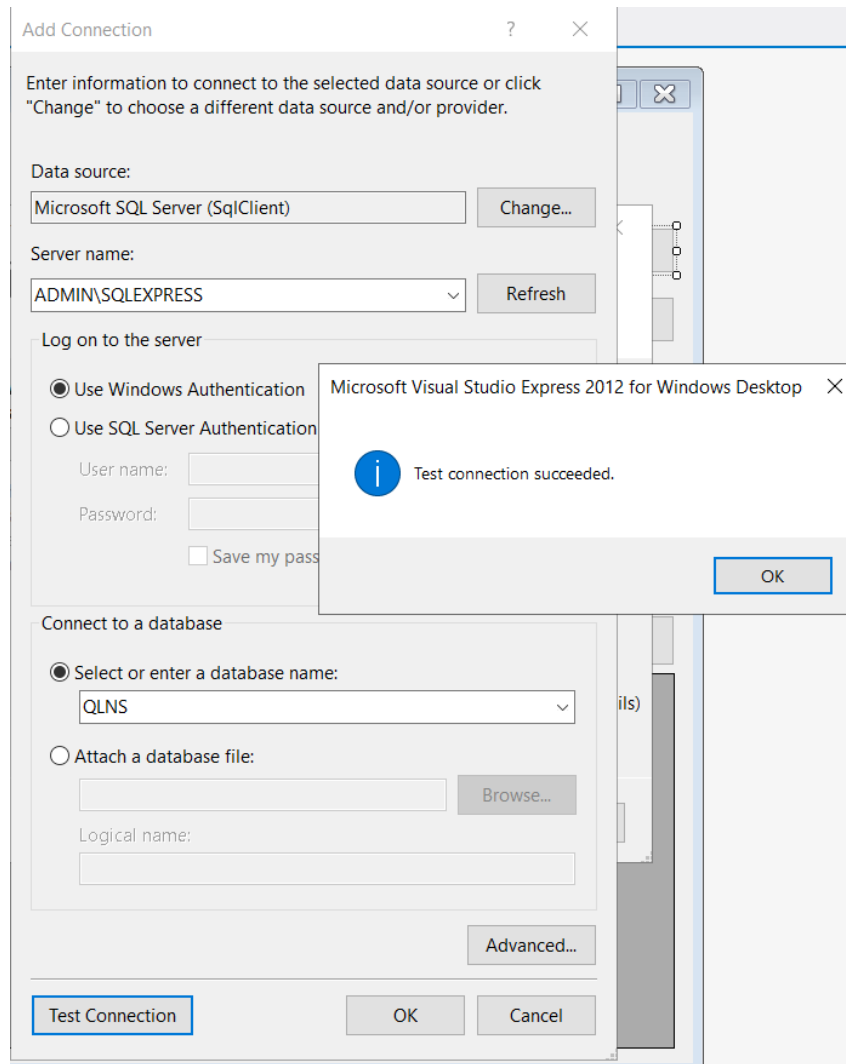
Cancel



- Chọn New Connection -> gõ vào server name: là tên server khi kết nối SQL Server.



- Chọn Database Name: QLNS
- Ấn Test Connection: Thành công là kết nối được



- Click vào Advanced -> Copy connection string ở Data Source:

Data Source=ADMIN\SQLEXPRESS;Initial Catalog=QLNS;Integrated Security=True

- Sau khi lấy được chuỗi kết nối -> Cancel hết các bước trên (Bước 3)

#### Bước 4.

- Sử dụng thư viện:

```
using System.Data.SqlClient;
```

- Tạo chuỗi kết nối mức Class: Với chuỗi kết nối đã được lấy ở bước 3, đặt dấu @ đầu chuỗi kết nối để kết nối tới CSDL.

```
SqlConnection conn;
```

```
string connstring = @"Data Source=ADMIN\SQLEXPRESS;Initial  
Catalog=QLNS;Integrated Security=True";
```

-----

- Sự kiện Form\_Load (frmEmployee\_Load):

- Mục đích khi bắt đầu Load Form hiển thị dữ liệu ban đầu như sau:
  - Hiển thị trên dgvEmployee danh sách thông tin các nhân viên.
  - Mặc định Gender: Giới tính = Nam
  - Ô nhập liệu tại txtMaNV
  - cboTenPB hiển thị danh sách các Phòng ban ở bảng Departments.

-----

```
private void frmEmployee_Load(object sender, EventArgs e)
{
    LayDuLieuDataGridView();
    LayDuLieuTenPB();
    //Mặc định giới tính là Nam
    radNam.Checked = true;
    radNu.Checked = false;
    //Ô nhập liệu tại MaNV
    txtMaNV.Focus();
}

private void LayDuLieuDataGridView()
{
    //Tạo chuỗi kết nối
    conn = new SqlConnection(connstring);
    //Mở kết nối
    conn.Open();
    //Câu lệnh lấy dữ liệu
    string sql = "select * from Employees";
    //bat dau truy van
    SqlCommand com = new SqlCommand(sql, conn);
    com.CommandType = CommandType.Text;
    //chuyen du lieu ve dataAdapter
    SqlDataAdapter da = new SqlDataAdapter(com);
    //tạo một kho ảo để lưu trữ dữ liệu
    DataTable dt = new DataTable();
    da.Fill(dt); // đổ dữ liệu vào kho
    da.Dispose(); //Giải phóng DataAdapter
    conn.Close(); // đóng kết nối
    dgvEmployee.DataSource = dt; //đổ dữ liệu vào datagridview
}
```

```

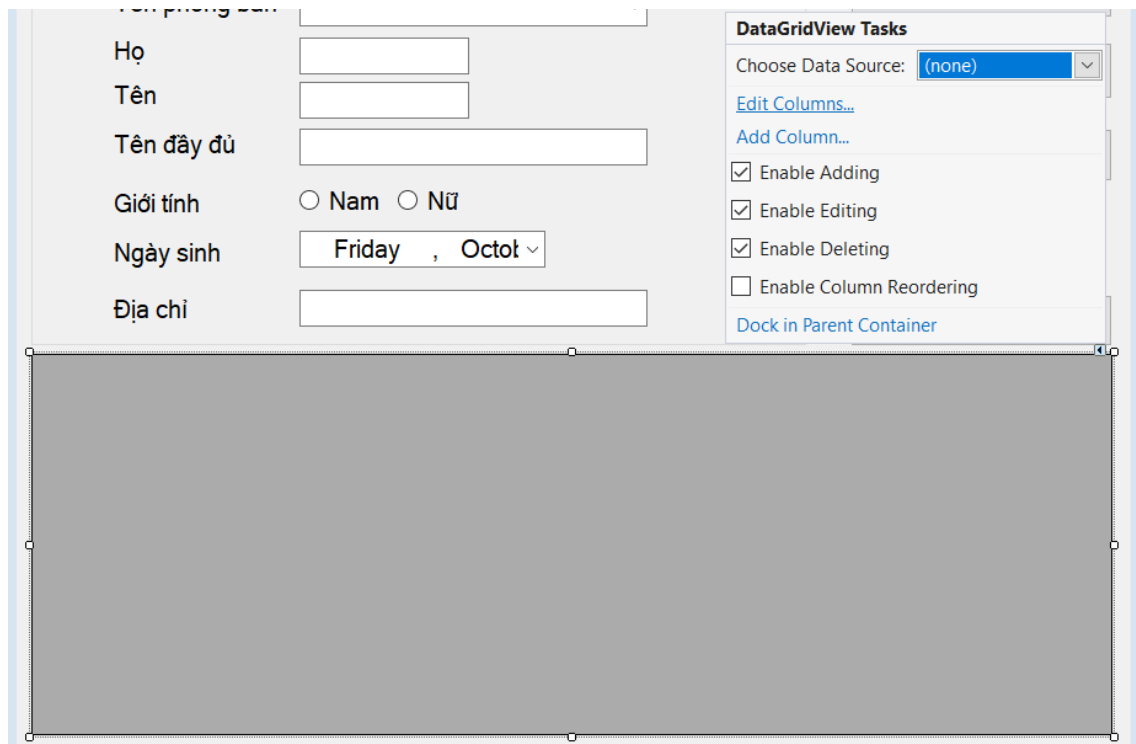
    }
private void LayDuLieuTenPB()
{
    //Tạo chuỗi kết nối
    conn = new SqlConnection(connstring);
    //Mở kết nối
    conn.Open();
    //Câu lệnh lấy dữ liệu
    string sql = "select DepartmentID,DepartmentName from
Departments";
    //bat dau truy van
    SqlCommand com = new SqlCommand(sql, conn);
    com.CommandType = CommandType.Text;
    //chuyen du lieu ve dataAdapter
    SqlDataAdapter da = new SqlDataAdapter(com);
    //tạo một kho ảo để lưu trữ dữ liệu
    DataTable dt = new DataTable();
    da.Fill(dt); // đổ dữ liệu vào kho
    da.Dispose(); //Giải phóng DataAdapter
    conn.Close(); // đóng kết nối
    //Gán dữ liệu nguồn cho comboBox
    cboTenPB.DataSource = dt;
    //Gán trường sẽ hiển thị trên ComboBox
    cboTenPB.DisplayMember = "DepartmentName";
    //Gán trường DepartmentID là mã ẩn sau mỗi DepartmentName
    //Vì trong bảng Employee chỉ có trường DepartmentID, cho nên
mỗi DepartmentName
    //sẽ tương ứng 1 DepartmentID truyền vào.
    cboTenPB.ValueMember = "DepartmentID";
}

```

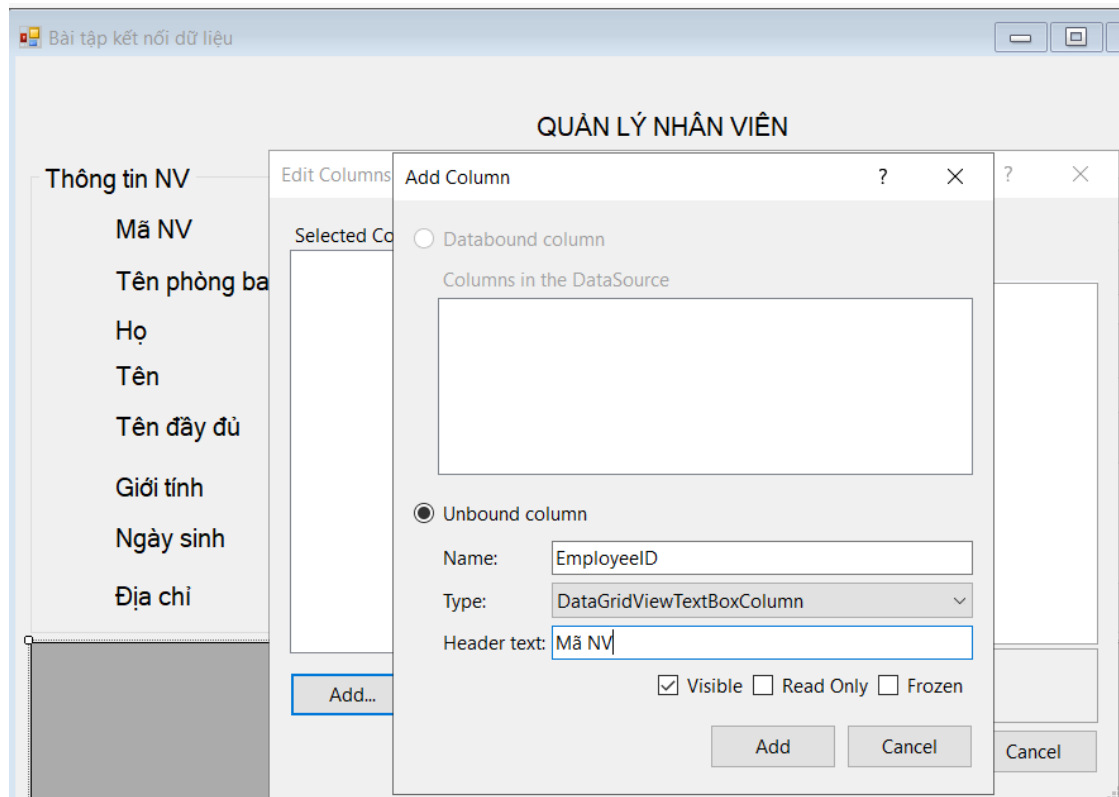
-----

#### Bước 5. Hiệu chỉnh hiển thị trên DataGridView:

- Do các hiển thị trên DataGridView là các mặc định columnName của bảng SQL Server, cho nên nhiều khi chúng ta cần hiển thị lại thông tin Header trên DataGridView cho dễ hiểu và đẹp hơn. Tạo ra các cột muốn hiển thị trên DataGridView và gán dữ liệu tương ứng. Bằng cách:



- Chọn Edit Columns -> Add:



- Giả sử chúng ta tạo được danh sách các Header như sau:



Edit Columns

Selected Columns:

- Mã NV
- Mã PB
- Họ
- Tên
- Tên đủ
- Giới tính
- Ngày sinh
- Địa chỉ

Add... Remove

Unbound Column Properties

|                  |                           |
|------------------|---------------------------|
| Visible          | True                      |
| <b>Behavior</b>  |                           |
| ContextMenuStrip | (none)                    |
| MaxInputLength   | 32767                     |
| ReadOnly         | False                     |
| Resizable        | True                      |
| SortMode         | Automatic                 |
| <b>Data</b>      |                           |
| DataPropertyName | (none)                    |
| <b>Design</b>    |                           |
| (Name)           | EmployeeID                |
| ColumnType       | DataGridViewTextBoxColumn |

**DataPropertyName**  
The name of the data source property or database ...

OK Cancel

- Ở trường: DataPropertyName của mỗi cột tương ứng chúng ta sẽ gõ thay thế chữ (none) -> tên cột tương ứng trong SQL Server của bảng này.

Edit Columns

Selected Columns:

- Mã NV
- Mã PB
- Họ
- Tên
- Tên đủ
- Giới tính
- Ngày sinh
- Địa chỉ

Add... Remove

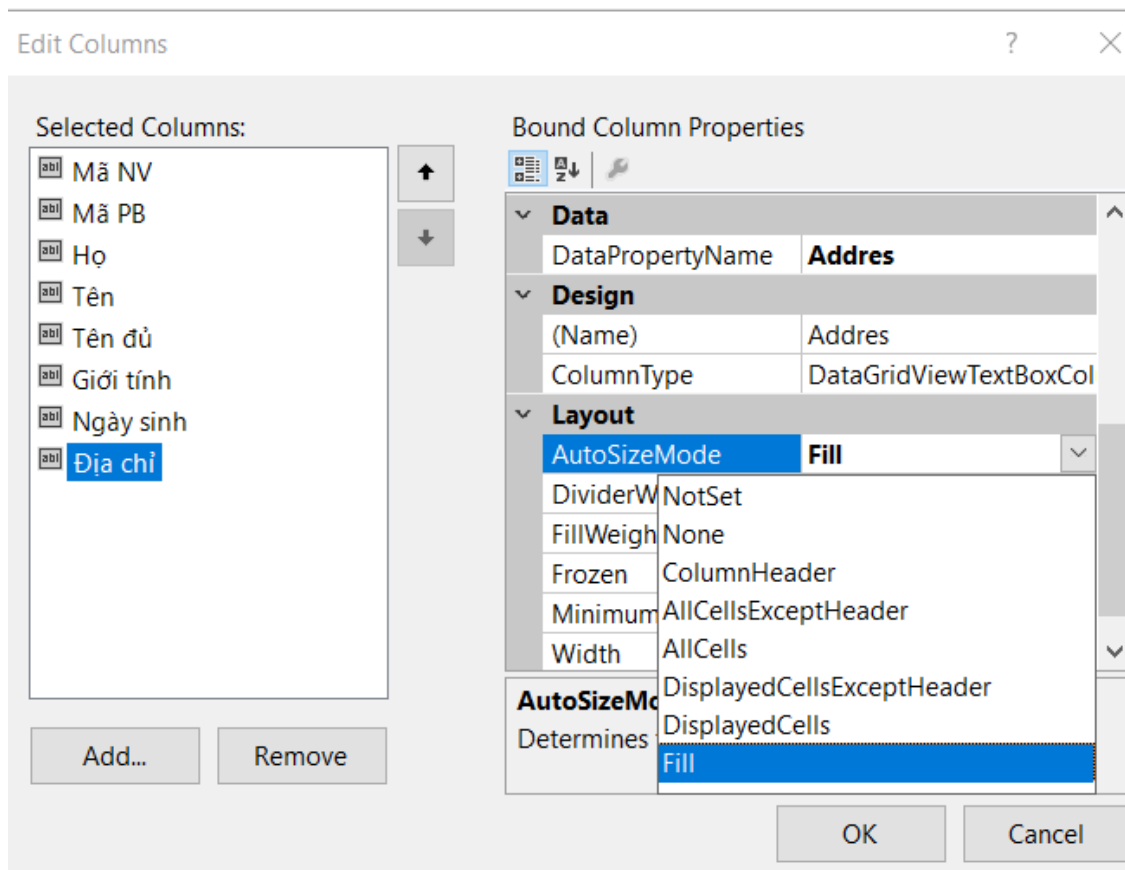
Bound Column Properties

|                  |                           |
|------------------|---------------------------|
| Visible          | True                      |
| <b>Behavior</b>  |                           |
| ContextMenuStrip | (none)                    |
| MaxInputLength   | 32767                     |
| ReadOnly         | False                     |
| Resizable        | True                      |
| SortMode         | Automatic                 |
| <b>Data</b>      |                           |
| DataPropertyName | EmployeeID                |
| <b>Design</b>    |                           |
| (Name)           | EmployeeID                |
| ColumnType       | DataGridViewTextBoxColumn |

**DataPropertyName**  
The name of the data source property or database ...

OK Cancel

- Một thao tác nữa là làm cho cột cuối cùng sẽ kéo đầy DataGridView. Để làm điều đó bạn hãy vào DataGridView -> Edit Columns -> Chọn cột Quê quán -> Chọn thuộc tính AutoSizeMode -> Chọn Fill.



Bước 6. Sự kiện thêm mới (btnAddNew\_Click):

- Kiểm tra các ô nhập liệu có rỗng hay không? Nếu thỏa mãn thì cho nhập -> Cập nhật lại DataGridView, ngược lại thông báo.

```
private void btnAddNew_Click(object sender, EventArgs e)
{
    if (txtMaNV.Text == "" || txtFirstName.Text == "" ||
    txtLastName.Text == "" || txtFullName.Text == "" || txtDiaChi.Text ==
    "" || cboTenPB.Text == "")
        MessageBox.Show("Bạn nhập thiếu thông tin!", "Thông
        báo");
    else
    {
        themMoiNhanVien();
        LayDuLieuDataGridView();
    }
}
```

```

    }
private void themMoiNhanVien()
{
    try
    {
        //Tạo chuỗi kết nối
        conn = new SqlConnection(connstring);
        //Mở kết nối
        conn.Open();
        //câu lệnh thêm mới bản ghi
        string strInsert = "insert into Employees values('" +
txtMaNV.Text + "',''" + cboTenPB.SelectedValue + "','N'";
        strInsert += txtFirstName.Text + "','N'" +
txtLastName.Text + "','N'" + txtFullName.Text + "','';
        //Kiểm tra giới tính
        string gt;
        if (radNam.Checked == true)
            gt = "Nam";
        else
            gt = "Nu";
        strInsert += gt + "',''" + dateTimePicker1.Value +
"'',N'" + txtDiaChi.Text + "')";
        //Khai báo và khởi tạo command
        SqlCommand cmd = new SqlCommand(strInsert, conn);
        //Thực thi câu lệnh delete
        cmd.ExecuteNonQuery();
        conn.Close();
    }
    catch
    {
        MessageBox.Show("Có lỗi nhập liệu!", "Thông báo");
    }
}
}

```

-----

Bước 7. Sửa dữ liệu thông tin nhân viên với manv không được sửa, nếu không có mã này thì thông báo không có nhân viên.

- Sự kiện btnEdit\_Click

```

private void btnEdit_Click(object sender, EventArgs e)
{
    if (KiemTraMaNV(int.Parse(txtMaNV.Text)) == false)
    {

```

```

        MessageBox.Show("Không có nhân viên này!", "Thông
báo");
    }
    else
    {
        if (txtFirstName.Text == "" || txtLastName.Text == ""
|| txtFullName.Text == "" || txtDiaChi.Text == "" || cboTenPB.Text ==
"")
            MessageBox.Show("Bạn nhập thiếu thông tin!",
"Thông báo");
        else
        {
            SuaNhanVien();
            LayDuLieuDataGridView();
        }
    }
}
private bool KiemTraMaNV(int manv)
{
    bool ktra = false;
    //Tạo chuỗi kết nối
    conn = new SqlConnection(connstring);
    //Mở kết nối
    conn.Open();
    //Câu lệnh lấy dữ liệu
    string sql = "select * from Employees where
EmployeeID='"+manv+"'";
    SqlCommand com = new SqlCommand(sql, conn); //bat dau truy
van
    com.CommandType = CommandType.Text;
    SqlDataAdapter da = new SqlDataAdapter(com); //chuyen du
lieu ve dataAdapter
    DataTable dt = new DataTable(); //tạo một kho ảo để lưu
trữ dữ liệu
    da.Fill(dt); // đổ dữ liệu vào kho
    da.Dispose(); //Giải phóng DataAdapter
    conn.Close(); // đóng kết nối
    if (dt.Rows.Count == 0)
        ktra = false;
    else
        ktra = true;
    return ktra;
}

```

```

private void SuaNhanVien()
{
    try
    {
        //Tạo chuỗi kết nối
        conn = new SqlConnection(connstring);
        //Mở kết nối
        conn.Open();
        //câu lệnh cập nhật bản ghi
        string strUpdate = "Update Employees set
DepartmentID='" + cboTenPB.SelectedValue + "',FirstName=N'";
        strUpdate += txtFirstName.Text + "',LastName=N'" +
txtLastName.Text + "',FullName=N'" + txtFullName.Text + "',Gender='";
        //Kiểm tra giới tính
        string gt;
        if (radNam.Checked == true)
            gt = "Nam";
        else
            gt = "Nu";
        strUpdate += gt + "',Birthday=' " +
dateTimePicker1.Value + "',Addres=N'" + txtDiaChi.Text + "' where
EmployeeID='"+txtMaNV.Text+"'";
        //Khai báo và khởi tạo command
        SqlCommand cmd = new SqlCommand(strUpdate, conn);
        //Thực thi câu lệnh delete
        cmd.ExecuteNonQuery();
        conn.Close();
    }
    catch
    {
        MessageBox.Show("Có lỗi nhập liệu!", "Thông báo");
    }
}

```

-----

Bước 8. Xóa nhân viên với manv nhập vào, nếu không có mã này thì thông báo không có nhân viên.

- Sự kiện btnDelete\_Click

```

private void btnDelete_Click(object sender, EventArgs e)
{
    if (KiemTraMaNV(int.Parse(txtMaNV.Text)) == false)
    {

```

```

        MessageBox.Show("Không có nhân viên này!", "Thông
báo");
    }
    else
    {
        XoaNhanVien();
        LayDuLieuDataGridView();
    }
}
private void XoaNhanVien()
{
    try
    {
        //Tạo chuỗi kết nối
        conn = new SqlConnection(connstring);
        //Mở kết nối
        conn.Open();
        //câu lệnh cập nhật bản ghi
        string strDelete = "Delete from Employees where
EmployeeID='" + txtMaNV.Text + "'";
        //Khai báo và khởi tạo command
        SqlCommand cmd = new SqlCommand(strDelete, conn);
        //Thực thi câu lệnh delete
        cmd.ExecuteNonQuery();
        conn.Close();
    }
    catch
    {
        MessageBox.Show("Có lỗi nhập liệu!", "Thông báo");
    }
}
}

```

-----

Bước 9. Nhập tiếp: sự kiện btnReset\_Click

```

private void btnReset_Click(object sender, EventArgs e)
{
    txtMaNV.Clear();
    LayDuLieuTenPB();
    txtFirstName.Clear();
    txtLastName.Clear();
    txtFullName.Clear();
    txtDiaChi.Clear();
    radNam.Checked = true;
}

```

```
radNu.Checked = false;  
txtMaNV.Focus();  
}
```

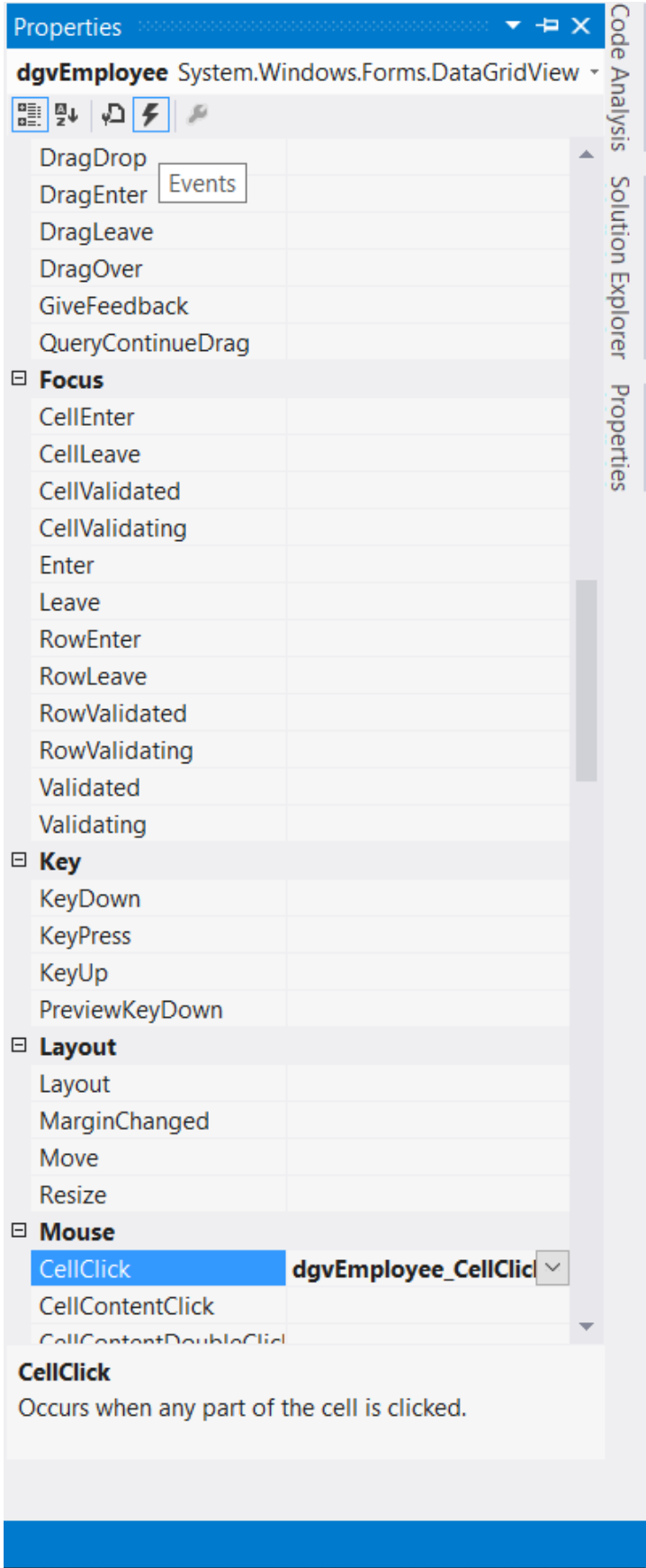
Bước 10. Thoát - Đóng chương trình - btnExit\_Click

```
private void btnExit_Click(object sender, EventArgs e)  
{  
    DialogResult ok = new DialogResult();  
    ok = MessageBox.Show("Bạn có muốn thoát?", "Question",  
MessageBoxButtons.YesNo, MessageBoxIcon.Question);  
    if (ok == DialogResult.Yes)  
        Application.Exit();  
}
```

-----

Bước 11. Click trên 1 dòng trên datagridview thì nội dung tương ứng sẽ hiển thị trên các control nhập liệu:

Bạn vào phần **event** trong **properties** của DataGridView và click đôi vào dòng **CellClick** để chuyển đến phần viết code cho sự kiện này.





```

private void dgvEmployee_CellClick(object sender,
DataGridViewCellEventArgs e)
{
    int numrow;
    numrow = e.RowIndex;
    txtMaNV.Text =
dgvEmployee.Rows[numrow].Cells[0].Value.ToString();
    cboTenPB.Text =
TenPB(int.Parse(dgvEmployee.Rows[numrow].Cells[1].Value.ToString()));
    txtFirstName.Text =
dgvEmployee.Rows[numrow].Cells[2].Value.ToString();
    txtLastName.Text =
dgvEmployee.Rows[numrow].Cells[3].Value.ToString();
    txtFullName.Text=
dgvEmployee.Rows[numrow].Cells[4].Value.ToString();
    if (dgvEmployee.Rows[numrow].Cells[5].Value.ToString() ==
"Nam")
    {
        radNam.Checked = true;
    }
    if (dgvEmployee.Rows[numrow].Cells[5].Value.ToString() ==
"Nu")
    {
        radNu.Checked = true;
    }
    dateTimePicker1.Text =
dgvEmployee.Rows[numrow].Cells[6].Value.ToString();
    txtDiaChi.Text =
dgvEmployee.Rows[numrow].Cells[7].Value.ToString();
}
private string TenPB(int MaPB)
{
    string tenphongban="";
    //Tạo chuỗi kết nối
    conn = new SqlConnection(connstring);
    //Mở kết nối
    conn.Open();
    //Câu lệnh lấy dữ liệu
    string sql = "select DepartmentName from Departments where
DepartmentID='"+MaPB+"'";
    SqlCommand com = new SqlCommand(sql, conn); //bat dau truy
van
    com.CommandType = CommandType.Text;

```

```
//sử dụng DataReader để đọc từng dòng dữ liệu
SqlDataReader dr = com.ExecuteReader();
while (dr.Read())
{
    tenphongban = dr["DepartmentName"].ToString();
}
dr.Close();
conn.Close();
return tenphongban;
}
```