

Chương 4: Giao diện người dùng (User Interface)

Gv: Đặng Hữu Nghị

Nội dung

4.1. Giới thiệu

4.1.1. Ứng dụng Windows Forms

4.1.2. Thanh công cụ (Toolbox)

4.2. Biểu mẫu (Form)

4.3. Các điều khiển thông thường

4.4. Các điều khiển đặc biệt

4.5. Điều khiển dùng để xây dựng menu

4.6. Điều khiển chứa các điều khiển khác

4.6. Điều khiển chứa các điều khiển khác

4.6.1. Điều khiển GroupBox

4.6.2. Điều khiển Panel

4.6.3. Điều khiển FlowLayoutPanel

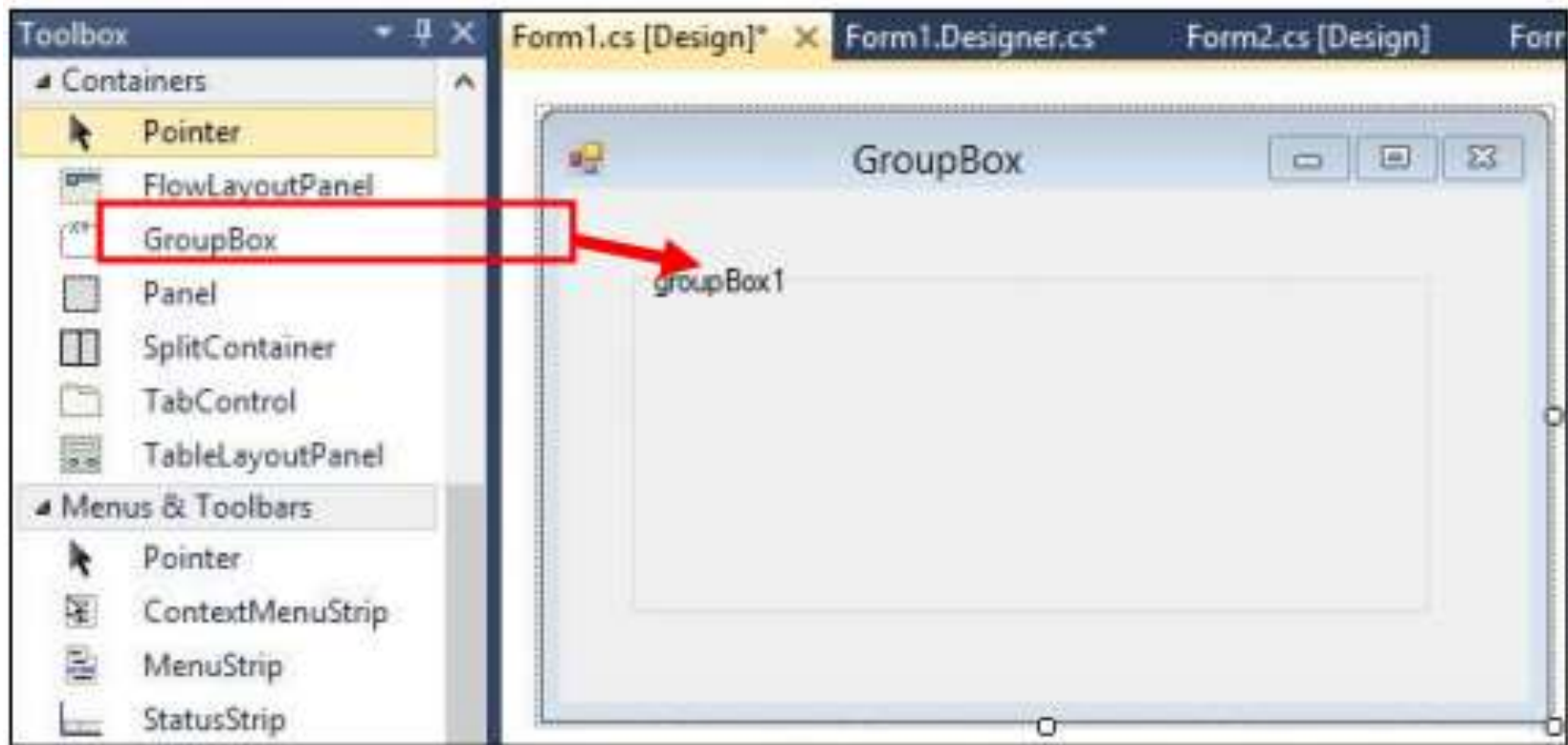
4.6.4. Điều khiển TableLayoutPanel

4.6.5. Điều khiển TabControl

4.6.1. Điều khiển GroupBox

- *GroupBox* là dạng điều khiển chứa, có thể chứa các điều khiển khác hiển thị trên form, giúp cho việc thiết kế giao diện của form dễ nhìn và khoa học hơn. *GroupBox* không hỗ trợ thanh trượt (ScrollBar).
- *GroupBox* có thể có tiêu đề hiển thị, tiêu đề này được thiết lập trong thuộc tính *Text*. Nếu không muốn hiển thị tiêu đề, lập trình viên có thể thiết lập chuỗi rỗng trong thuộc tính *Text*. Điều khiển *GroupBox* được đặt trong nhóm Containers của cửa sổ Toolbox

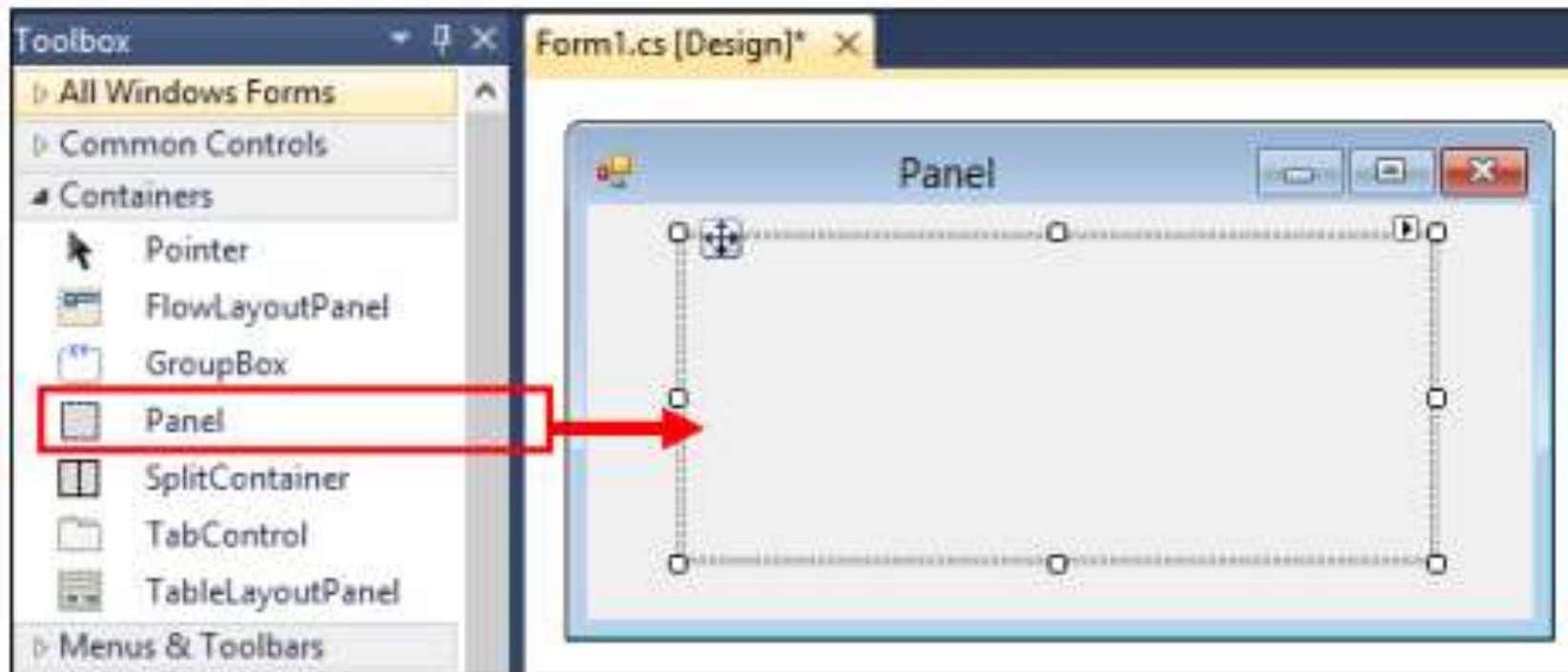
4.6.1. Điều khiển GroupBox



4.6.2. Điều khiển Panel

- Cũng như GroupBox, Panel là một điều khiển dùng để chứa các điều khiển khác. Panel có các thuộc tính AutoSize, AutoSizeMode như GroupBox và thuộc tính đường viền BorderStyle như Label.
- Điểm khác biệt của Panel với GroupBox là điều khiển Panel không có tiêu đề mô tả (không có thuộc tính Text) và có thanh trượt ScrollBar ngang và ScrollBar dọc (có thuộc tính AutoScroll).

4.6.2. Điều khiển Panel



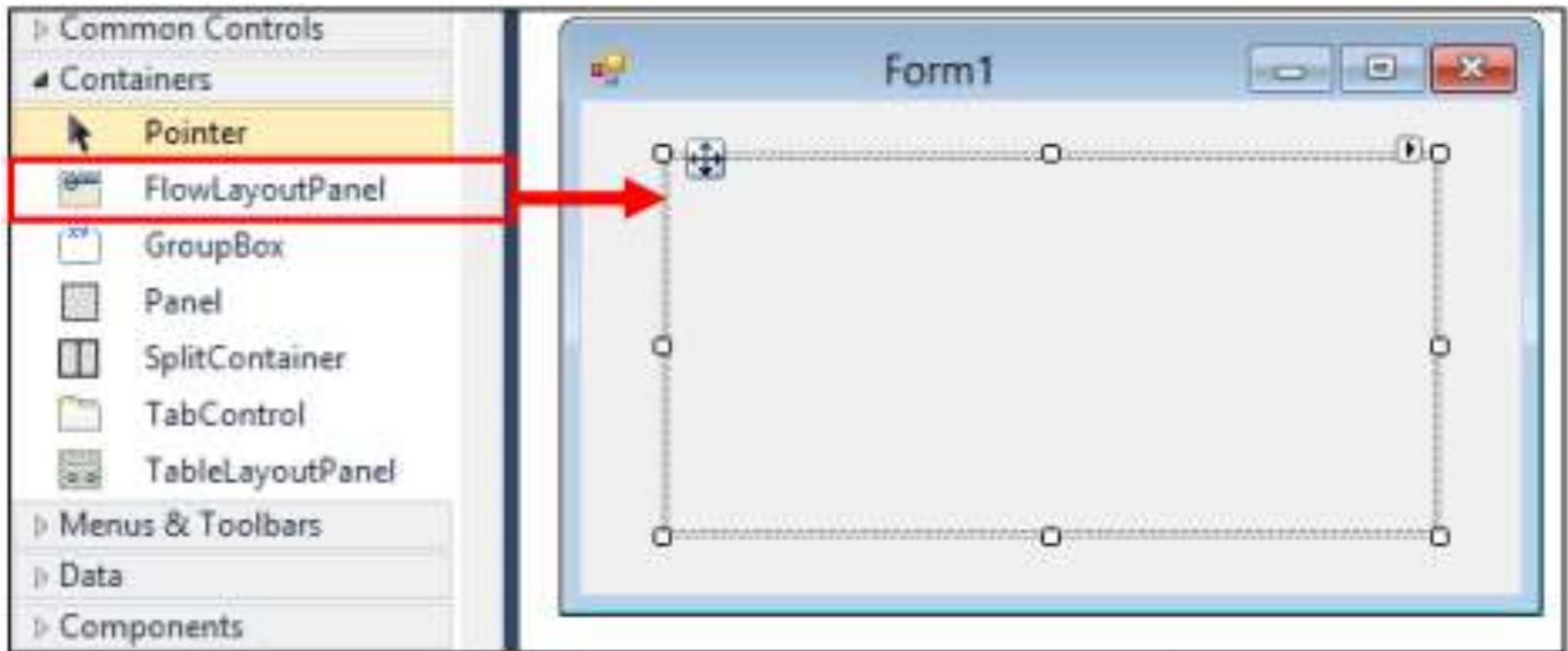
4.6.2. Điều khiển Panel

Thuộc tính	Mô tả
<i>AutoScroll</i>	<p>Mang giá trị True hoặc False.</p> <ul style="list-style-type: none">- Nếu là True: <i>Panel</i> tự động xuất hiện thanh trượt khi kích thước Panel không thể hiển thị hết các điều khiển chứa bên trong- Nếu là False: Panel sẽ không hiển thị thanh trượt
<i>BorderStyle</i>	<p>Kiểu đường viền của Panel khi hiển thị. Có 3 giá trị: None, FixedSingle và Fixed3D</p> <ul style="list-style-type: none">- None: Không hiển thị đường viền- FixedSingle: Quanh Panel sẽ hiển thị một đường viền đơn.- Fixed3D: Hiển thị đường viền của Panel dạng 3 chiều.

4.6.3. Điều khiển FlowLayoutPanel

- *FlowLayoutPanel* là lớp con của điều khiển *Panel*, do đó có thể chứa các điều khiển khác như *Panel*.
- Mục đích chính của *FlowLayoutPanel* là giúp bố trí các điều khiển trên form một cách có tổ chức và khoa học.
- Khi thêm một điều khiển nào đó vào *FlowLayoutPanel* thì *FlowLayoutPanel* sẽ tự động sắp xếp các điều khiển đặt bên trong theo quy tắc định trước và đồng thời cũng thay đổi kích thước của các điều khiển bên trong cho phù hợp với kích thước của *FlowLayoutPanel*.

4.6.3. Điều khiển FlowLayoutPanel



4.6.3. Điều khiển *FlowLayoutPanel*

- Điều khiển *FlowLayoutPanel* cũng hỗ trợ dạng thanh trượt (*ScrollBar*) như *Panel*, khi thuộc tính *AutoScroll* được thiết lập là *True* thì khi kích thước các điều khiển được chứa vượt ngoài kích thước *FlowLayoutPanel*, thì *FlowLayoutPanel* sẽ hiển thị thanh trượt.
- Việc bố trí các điều khiển khi thêm vào *FlowLayoutPanel* như thế nào là do thuộc tính *FlowDirection* quy định. Thuộc tính này mang 4 giá trị cho phép các điều khiển lần lượt thêm vào theo 4 hướng: từ trái qua phải (*LeftToRight*), từ phải qua trái (*RightToLeft*), từ trên xuống (*TopDown*) và từ dưới lên (*BottomUp*).

4.6.3. Điều khiển FlowLayoutPanel

- Các điều khiển được thêm vào đến khi vượt ngoài phạm vi của *FlowLayoutPanel*, nếu muốn các điều khiển tự động bố trí xuống dòng mới hoặc sang một cột mới thì cần phải thiết lập thuộc tính *WrapContents* là True. Còn nếu thuộc tính *WrapContents* là False thì *FlowLayoutPanel* sẽ hiển thị thanh trượt (thuộc tính *AutoScroll* là True) để hiển thị các điều khiển nằm ngoài phạm vi.

4.6.3. Điều khiển FlowLayoutPanel

- Thuộc tính *FlowDirection* là *LeftToRight*



- Thuộc tính *FlowDirection* là *RightToLeft*.

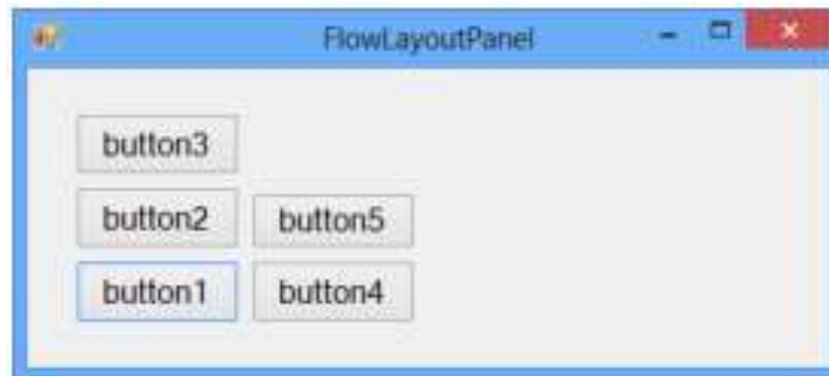


4.6.3. Điều khiển FlowLayoutPanel

- Thuộc tính *FlowDirection* là *TopDown*:



- Thuộc tính *FlowDirection* là *BottomUp*



4.6.3. Điều khiển FlowLayoutPanel

Thuộc tính	Mô tả
<i>AutoScroll</i>	Mang giá trị True hoặc False. <ul style="list-style-type: none">- Nếu là True: <i>FlowLayoutPanel</i> tự động xuất hiện thanh trượt khi kích thước Panel không thể hiển thị hết các điều khiển chứa bên trong- Nếu là False: <i>FlowLayoutPanel</i> sẽ không hiển thị thanh trượt
<i>BorderStyle</i>	Kiểu đường viền của <i>FlowLayoutPanel</i> khi hiển thị. Có 3 giá trị: None, FixedSingle và Fixed3D <ul style="list-style-type: none">- None: Không hiển thị đường viền- FixedSingle:Quanh <i>FlowLayoutPanel</i> sẽ hiển thị một đường viền đơn.- Fixed3D: Hiển thị đường viền của <i>LowLayoutPanel</i> dạng 3 chiều.
<i>FlowDirection</i>	Cách thức bố trí các điều khiển khi các điều khiển nằm ngoài phạm vi của <i>FlowLayoutPanel</i> . Bao gồm 4 giá trị: <i>eftToRight</i> , <i>RightToLeft</i> , <i>TopDown</i> , <i>BottomUp</i>

4.6.3. Điều khiển FlowLayoutPanel

Thuộc tính	Mô tả
<i>WrapContents</i>	<p>Mang giá trị True hoặc False.</p> <ul style="list-style-type: none">- Nếu là True: Các điều khiển vượt ngoài kích thước <i>FlowLayoutPanel</i> sẽ tự động bố trí trên một dòng mới hoặc một cột mới- Nếu là False: <i>FlowLayoutPanel</i> sẽ xuất hiện thanh trượt để hiển thị các điều khiển ngoài kích thước của <i>FlowLayoutPanel</i> (với thuộc tính <i>AutoScroll</i> là True). Nếu thuộc tính <i>AutoScroll</i> là False thì các điều khiển nằm ngoài kích thước sẽ bị ẩn đi.

Phương thức	Mô tả
<i>Controls.Add</i>	Phương thức có chức năng thêm điều khiển vào <i>FlowLayoutPanel</i>

4.6.3. Điều khiển FlowLayoutPanel

- Ví dụ: Thiết kế giao diện chương trình gồm có 1 PictureBox và 1 FlowLayoutPanel như hình. Với FlowLayoutPanel chứa danh sách các hình. Khi người dùng nhấp chuột chọn hình nào trong FlowLayoutPanel thì hình đó sẽ hiển thị trên PictureBox.



4.6.3. Điều khiển FlowLayoutPanel

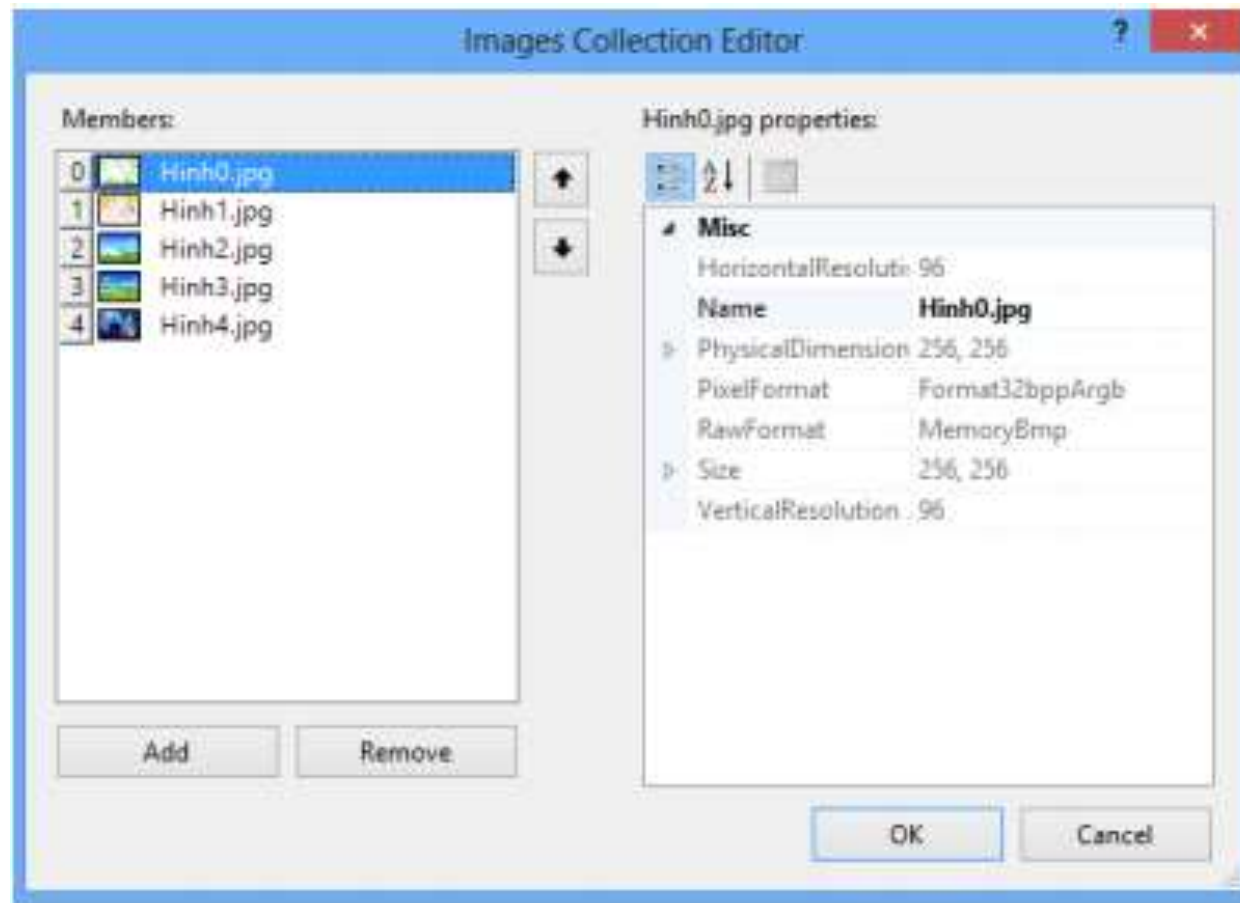
- Bước 1: Thiết kế giao diện chương trình. Kéo các điều khiển FlowLayoutPanel, PictureBox và ImageList và form như hình



4.6.3. Điều khiển FlowLayoutPanel

- Bước 2: Thiết lập giá trị thuộc tính cho điều khiển trong cửa sổ Properties
 - ✓ Form1:
 - Thuộc tính Text: “FlowLayoutPanel”
 - ✓ pictureBox1:
 - Thuộc tính Name: myPictureBox
 - ✓ flowLayoutPanel1:
 - Thuộc tính Name: myFlowLayoutPanel
 - Thuộc tính BorderStyle: FixSingle
 - Thuộc tính AutoScroll: True
 - Thuộc tính WrapContents: False
 - ✓ imageList1:
 - Thuộc tính Name: myImageList
 - Thuộc tính ImageSize: 256, 256
 - Thuộc tính Images: Thêm một số hình ảnh vào myImageList như hình

4.6.3. Điều khiển FlowLayoutPanel



4.6.3. Điều khiển FlowLayoutPanel

- Bước 3: Viết mã lệnh cho điều khiển
- ✓ Sự kiện *Load* của Form1:

```
private void Form1_Load(object sender, EventArgs e)
{
    for (int i = 0; i < myImageList.Images.Count; i++)
    {
        PictureBox pic = new PictureBox();
        pic.Image = myImageList.Images[i];
        pic.Size = new Size(50, 50);
        pic.Click += new EventHandler(Form1_Click);
        myFlowLayoutPanel.Controls.Add(pic);
    }
}
```

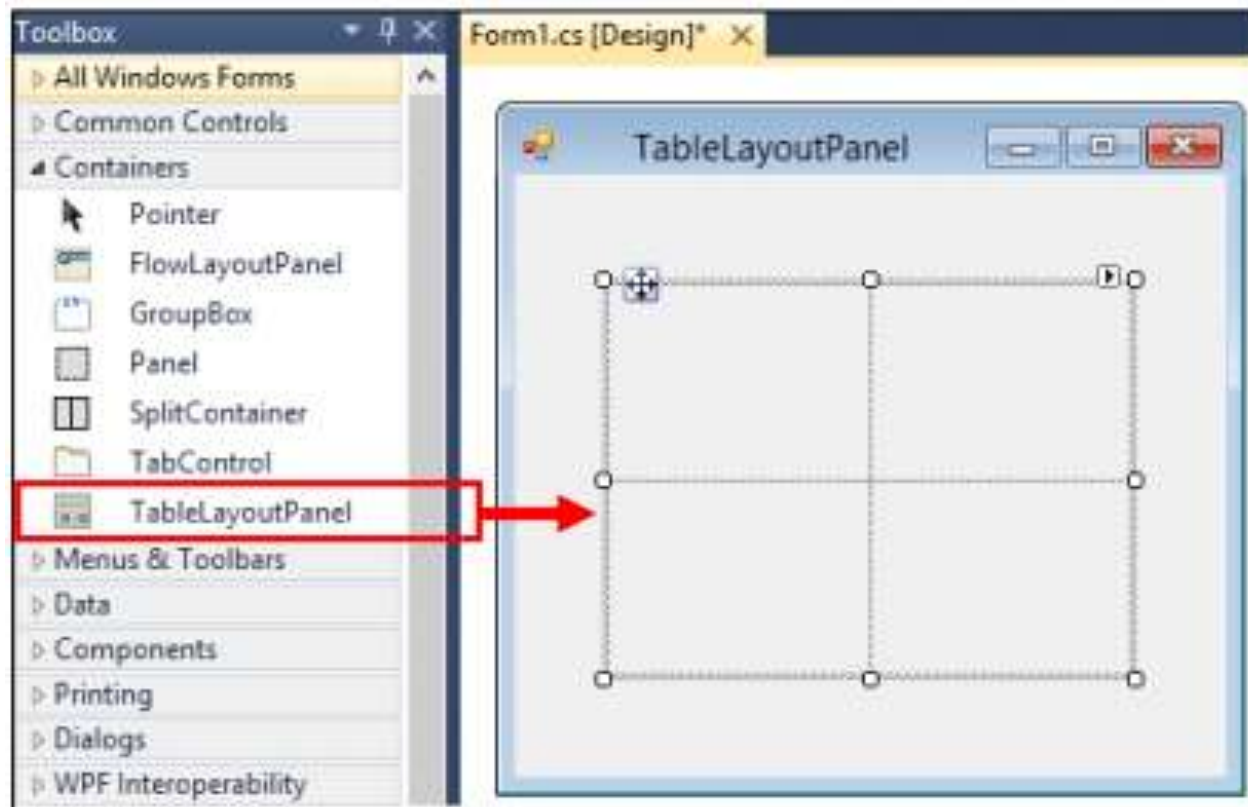
4.6.3. Điều khiển FlowLayoutPanel

✓ Sự kiện *Click* của Form1:

```
private void Form1_Click(object sender, EventArgs e)
{
    try
    {
        PictureBox pic = (PictureBox)sender;
        myPictureBox.Image = pic.Image;
    }
    catch (Exception ex) { }
}
```

4.6.4. Điều khiển *TableLayoutPanel*

- Cũng như điều khiển *FlowLayoutPanel*, *TableLayoutPanel* là điều khiển dẫn xuất từ điều khiển *Panel* và được dùng cho mục đích thiết kế giao diện form

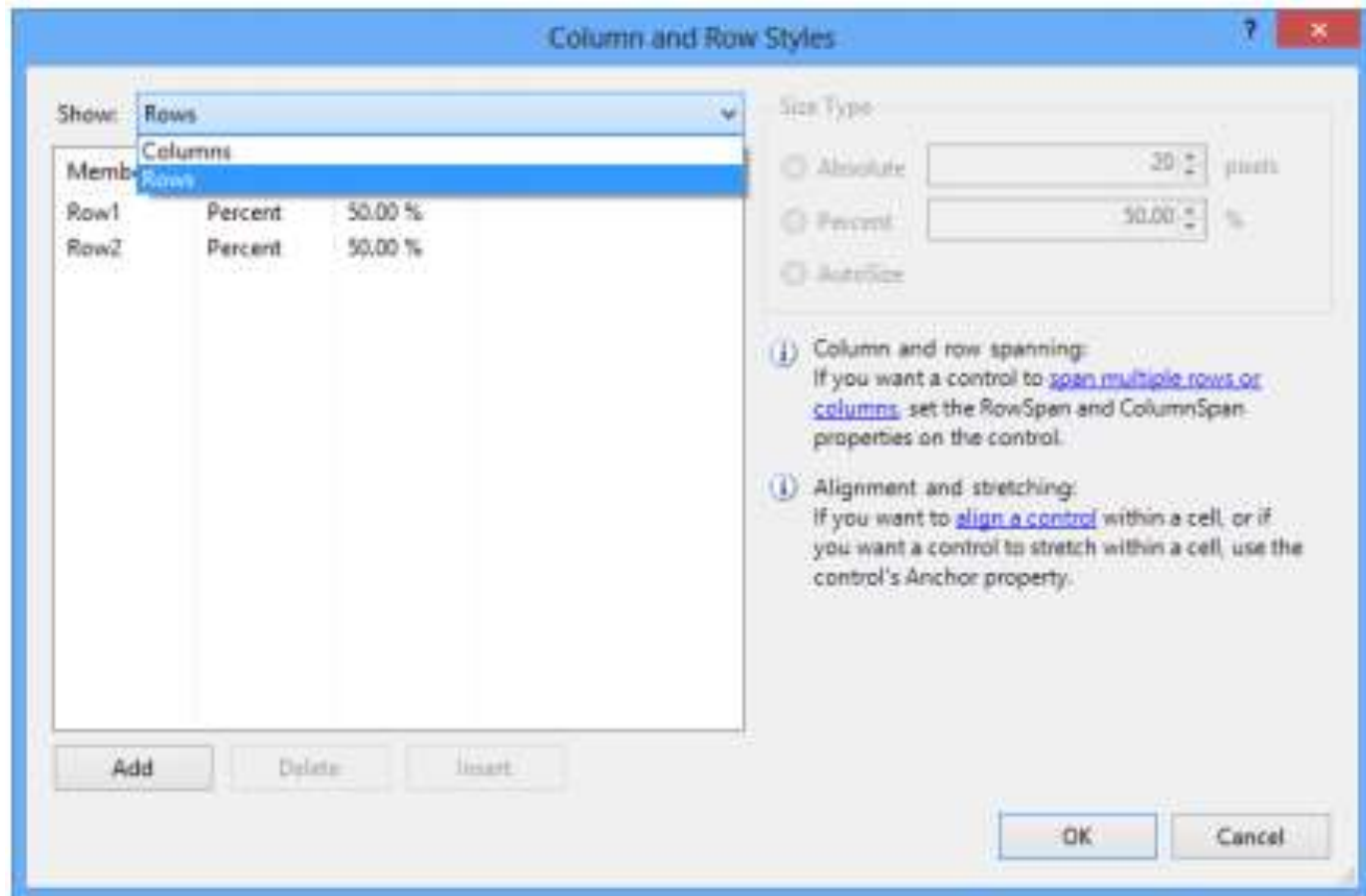


4.6.4. Điều khiển *TableLayoutPanel*

- *TableLayoutPanel* bao gồm các ô theo dòng và cột để thêm điều khiển vào.
- Ta có thể thêm các dòng và các cột cho *TableLayoutPanel* qua các thuộc tính *Columns* và *Rows* trong cửa sổ Properties, hoặc thêm dòng và cột qua *ContextMenuStrip* khi nhấp chuột phải vào *TableLayoutPanel*

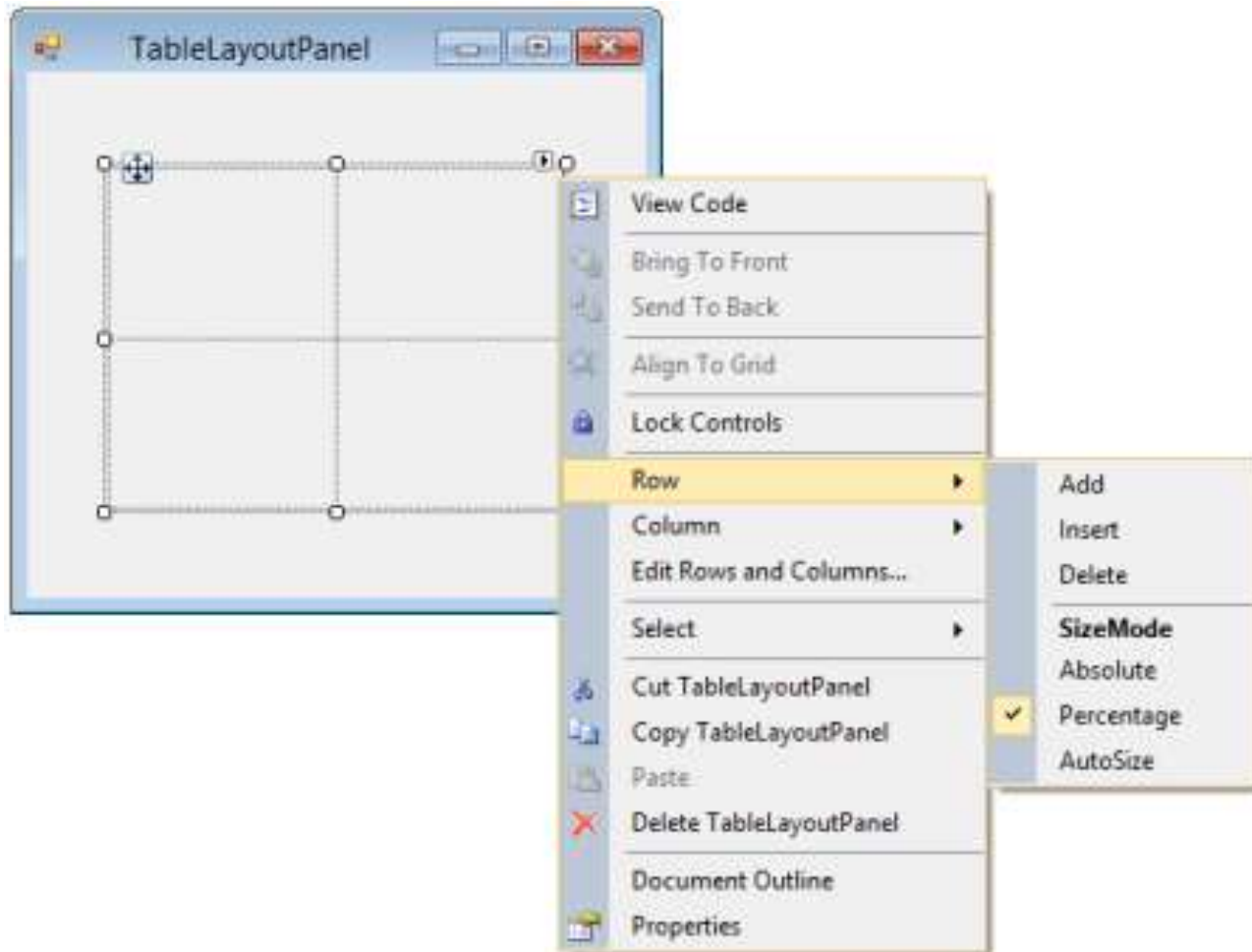
4.6.4. Điều khiển TableLayoutPanel

- *Thêm dòng và cột trên cửa sổ Column and Row Styles*



4.6.4. Điều khiển TableLayoutPanel

- *Thêm dòng hoặc cột trên ContextMenuStrip*



4.6.4. Điều khiển *TableLayoutPanel*

- Với mỗi ô của *TableLayoutPanel*, chỉ có thể chứa được một điều khiển. Tuy nhiên ta có thể thêm nhiều điều khiển trong một ô bằng cách thêm một điều khiển loại Containers như: *GroupBox*, *Panel*, ... vào ô của *TableLayoutPanel*, khi đó ta có thể thêm nhiều điều khiển vào điều khiển loại Containers nằm trong ô của *TableLayoutPanel*.
- *TableLayoutPanel* cũng được cung cấp thanh trượt (*ScrollBar*) khi thuộc tính *AutoScroll* là *True*.
- *TableLayoutPanel* là điều khiển dạng bảng chia thành các ô (cell) do đó không có thuộc tính *BorderStyle* mà thay vào đó có thuộc tính *CellBorderStyle*. Thuộc tính *CellBorderStyle* chỉ định kiểu đường viền cho *TableLayoutPanel*. Thông thường, thuộc tính *TableLayoutStyle* có giá trị mặc định là *None*, nghĩa là không hiển thị đường viền quanh các ô của *TableLayoutPanel*. Ta có thể thiết lập các loại đường viền được hỗ trợ cho thuộc tính *CellBorderStyle* như: *Single*, *Inset*, *InsetDouble*, *Outset*, *OutsetDouble*, hoặc *OutsetPartial*.

4.6.4. Điều khiển *TableLayoutPanel*

- Một số thuộc tính thường dùng của *TableLayoutStyle*

Thuộc tính	Mô tả
<i>AutoScroll</i>	Mang giá trị True hoặc False. <ul style="list-style-type: none">- Nếu là True: <i>TableLayoutStyle</i> tự động xuất hiện thanh trượt khi kích thước Panel không thể hiển thị hết các điều khiển chứa bên trong- Nếu là False: <i>TableLayoutStyle</i> sẽ không hiển thị thanh trượt
<i>ColumnCount</i>	Số cột của <i>TableLayoutPanel</i> . Lập trình viên có thể thêm hoặc giảm số cột bằng cách thay đổi giá trị thuộc tính <i>ColumnCount</i>
<i>Columns</i>	Thuộc tính này giúp hiển thị bảng Column and Rown Styles để thêm, sửa hoặc xóa cột.
<i>RowCount</i>	Số cột của <i>TableLayoutPanel</i> . Ta có thể thêm hoặc giảm số dòng bằng cách thay đổi giá trị thuộc tính <i>RowCount</i>
<i>Rows</i>	Thuộc tính này giúp hiển thị bảng Column and Rown Styles để thêm, sửa hoặc xóa dòng.

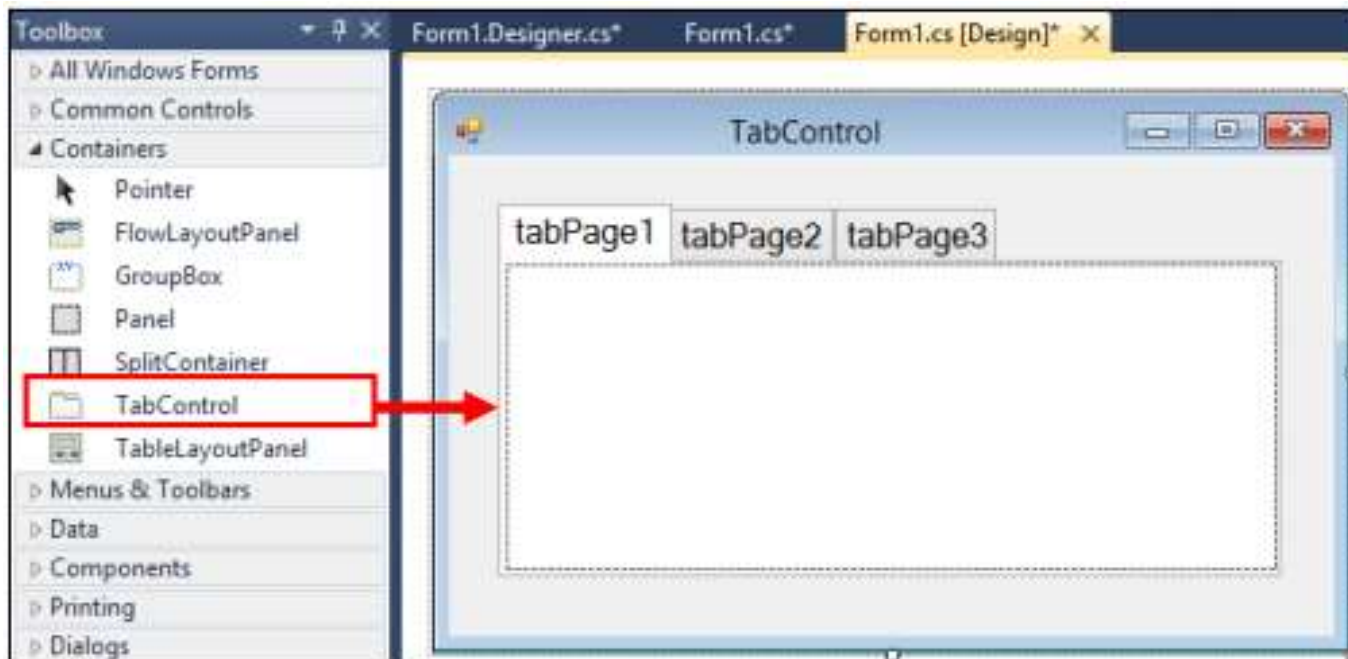
4.6.4. Điều khiển TableLayoutPanel

- Phương thức thường dùng của *TableLayoutStyle*

Phương thức	Mô tả
<i>Controls.Add</i>	Phương thức có chức năng thêm điều khiển vào <i>TableLayoutStyle</i>

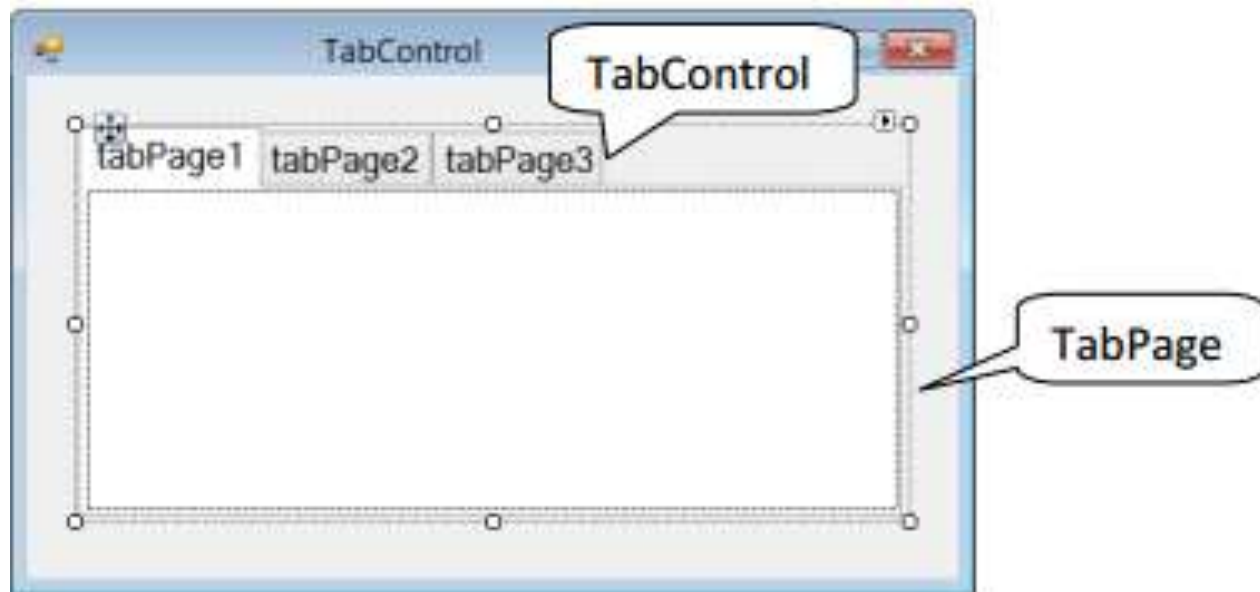
4.6.5. Điều khiển TabControl

- TabControl là điều khiển dạng Containers, do đó có thể chứa các điều khiển khác.
- Điểm đặc biệt của TabControl là cho phép thể hiện nhiều page trên một form duy nhất.
- Mỗi page có thể chứa nhiều điều khiển khác bên trong



4.6.5. Điều khiển TabControl

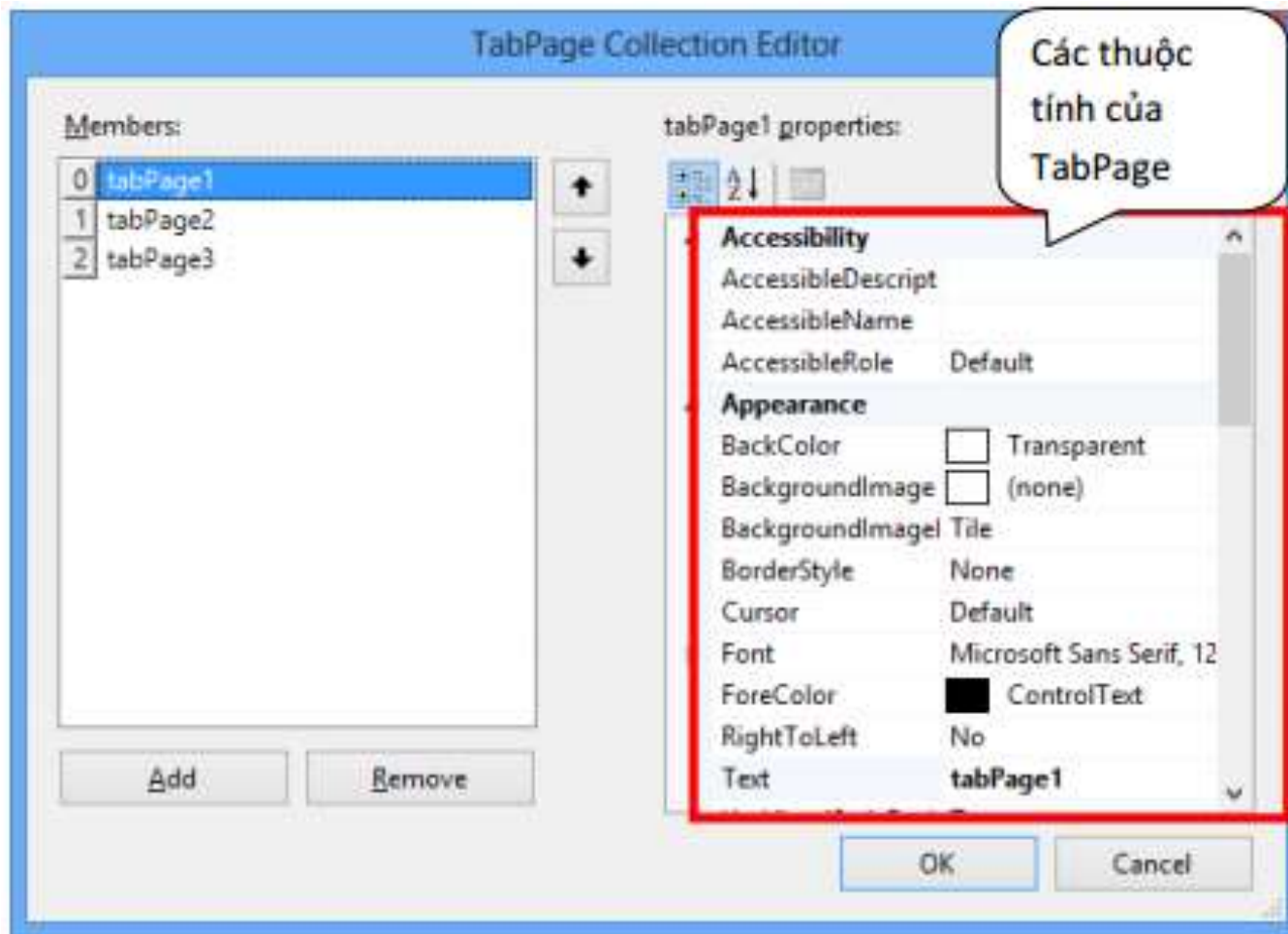
- TabPage:
 - ✓ Thuộc tính quan trọng nhất của *TabControl* là *TabPage*. Một *TabControl* có thể có nhiều *TabPage* như hình. Người dùng có thể nhấp vào các tab để chuyển đổi qua lại giữa các *TabPage* với nhau



4.6.5. Điều khiển TabControl

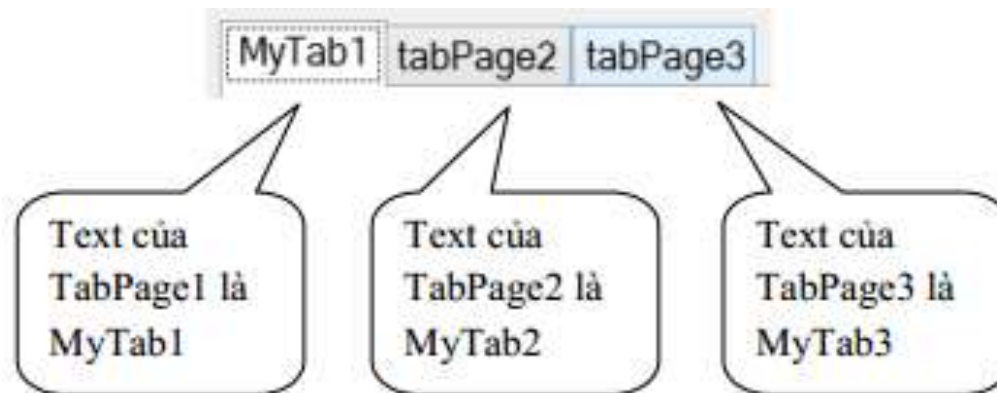
- ✓ *TabPage* là điều khiển dạng Container nằm trong *TabControl* và có thể chứa các điều khiển khác bên trong.
- ✓ Mỗi *TabPage* có các thuộc tính riêng, ta có thể thiết lập giá trị thuộc tính khác nhau trên mỗi *TabPage* của *TabControl* bằng cách nhấp chuột trái chọn thuộc tính *TabPage*s trên cửa sổ Properties. Khi đó một cửa sổ *TabPage Collection Editor* sẽ hiển thị như hình sau.
- ✓ Tại cửa sổ này, ta cũng có thể thêm hoặc xóa các *TabPage* bằng cách nhấn nút Add hoặc Remove.

4.6.5. Điều khiển TabControl



4.6.5. Điều khiển TabControl

- ✓ Điều khiển *TabPage* có nhiều điểm giống với điều khiển *Panel*. *TabPage* cũng hỗ trợ thanh trượt khi cần nếu như thuộc tính *AutoScroll* được thiết lập là *True*, có thuộc tính *BorderStyle* để thiết lập đường viền quanh *TabPage* với 3 giá trị: *None*, *FixedSingle*, *Fixed3D*.
- ✓ Tuy nhiên có điểm khác biệt với *Panel* là *TabPage* hỗ trợ thuộc tính *Text*, chuỗi mô tả được thiết lập trong thuộc tính *Text* sẽ hiển thị trên tab của *TabPage*:



4.6.5. Điều khiển TabControl

Các thuộc tính thường dùng của *TabControl*

Thuộc tính	Mô tả
<i>Appearance</i>	Thuộc tính chỉ định <i>TabPage</i> sẽ hiển thị ở hình dạng nào. Có 3 giá trị: <i>Normal</i> , <i>Button</i> , <i>FlatButtons</i>
<i>Alignment</i>	Thuộc tính xác định các tab sẽ hiển thị ở trên, dưới, trái hay phải của <i>TabControl</i> . Gồm các giá trị: <i>Top</i> , <i>Bottom</i> , <i>Left</i> , <i>Right</i>
<i>Multiline</i>	Mang hai giá trị True hoặc False. <ul style="list-style-type: none">- Nếu là True: Cho phép hiển thị nhiều dòng để chứa các tab nếu số lượng các tab vượt ngoài phạm vi kích thước của <i>TabControl</i>.- Nếu là False: Chỉ cho phép tab hiển thị trên một dòng.
<i>TabPage</i>	Chứa tập các các <i>TabPage</i> có trong <i>TabControl</i>
<i>TabCount</i>	Trả về số lượng <i>TabPage</i> mà <i>TabControl</i> có
<i>SelectedTab</i>	Trả về điều khiển <i>TabPage</i> được chọn
<i>SelectedIndex</i>	Trả về vị trí của <i>TabPage</i> được chọn

4.6.5. Điều khiển TabControl

- Các sự kiện thường dùng của *TabControl*

Sự kiện	Mô tả
<i>SelectedIndexChanged</i>	Phát sinh khi người dùng chọn một <i>TabPage</i> khác trên <i>TabControl</i>

4.6.5. Điều khiển TabControl

- Ví dụ: Viết chương trình quản lý nhân sự như hình. Chương trình gồm 2 *TabPage*: *TabPage* Quản lý nhân viên và *TabPage* Quản lý giáo viên.
 - ✓ *TabPage* quản lý nhân viên: Cho phép thêm, sửa và xóa nhân viên. Thông tin nhân viên cần quản lý bao gồm: Họ tên nhân viên, chức vụ của nhân viên, hệ số lương và lương cơ bản.
 - ✓ *TabPage* Quản lý giáo viên: Cho phép thêm sửa xóa giáo viên. Thông tin giáo viên cần quản lý gồm: Họ tên giáo viên, chức vụ của giáo viên, tiền giảng một tiết, số tiết dạy và học vị của giáo viên.

4.6.5. Điều khiển TabControl

The screenshot shows a Windows application titled 'TabControl'. The main window has a tab control with two tabs: 'Quản lý nhân viên' (selected) and 'Quản lý giáo viên' (Teachers Management). The 'Quản lý nhân viên' tab contains the following elements:

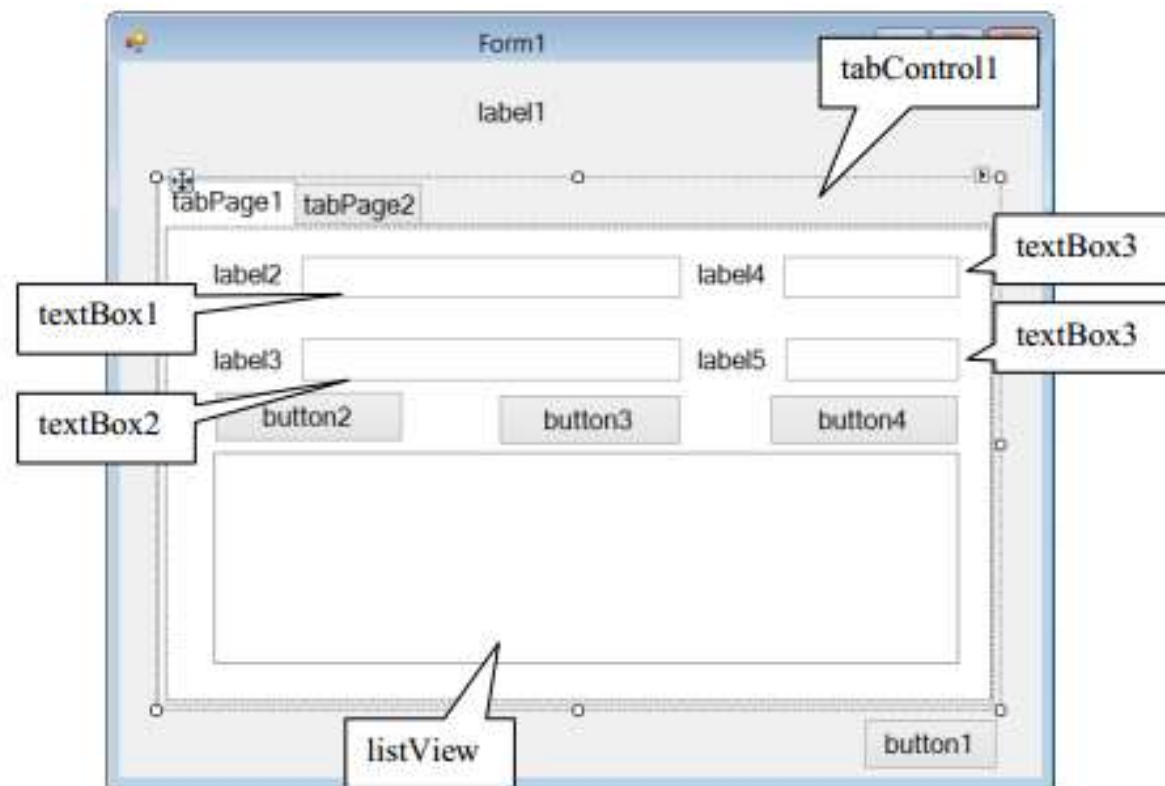
- Form fields: 'Họ tên:' (Last Name), 'Chức vụ:' (Position), 'Lương cơ bản:' (Basic Salary), and 'Hệ số lương:' (Salary Coefficient).
- Buttons: 'Thêm nhân viên' (Add Employee), 'Cập nhật' (Update), and 'Xóa nhân viên' (Delete Employee).
- Table: A table with columns 'Họ tên', 'Chức vụ', 'Hệ số lương', and 'Lương cơ bản'.
- Bottom button: 'Thoát' (Exit).

The screenshot shows the same 'TabControl' application, but with the 'Quản lý giáo viên' tab selected. The 'Quản lý nhân viên' tab is now disabled. The 'Quản lý giáo viên' tab contains the following elements:

- Form fields: 'Họ tên:' (Last Name), 'Học vị:' (Degree), 'Tiền dạy một tiết:' (Fee per lesson), and 'Số tiết dạy:' (Number of lessons).
- Buttons: 'Thêm giáo viên' (Add Teacher), 'Cập nhật' (Update), and 'Xóa giáo viên' (Delete Teacher).
- Table: A table with columns 'Họ tên', 'Học vị', 'Tiền dạy một tiết', and 'Số tiết dạy'.
- Bottom button: 'Thoát' (Exit).

4.6.5. Điều khiển TabControl

- Tạo *TabPage* Quản lý nhân viên:
 - ✓ Bước 1: Thiết kế giao diện ban đầu. Thêm các điều khiển *Label*, *TextBox*, *TabControl* và *ListView* vào form như hình



4.6.5. Điều khiển TabControl

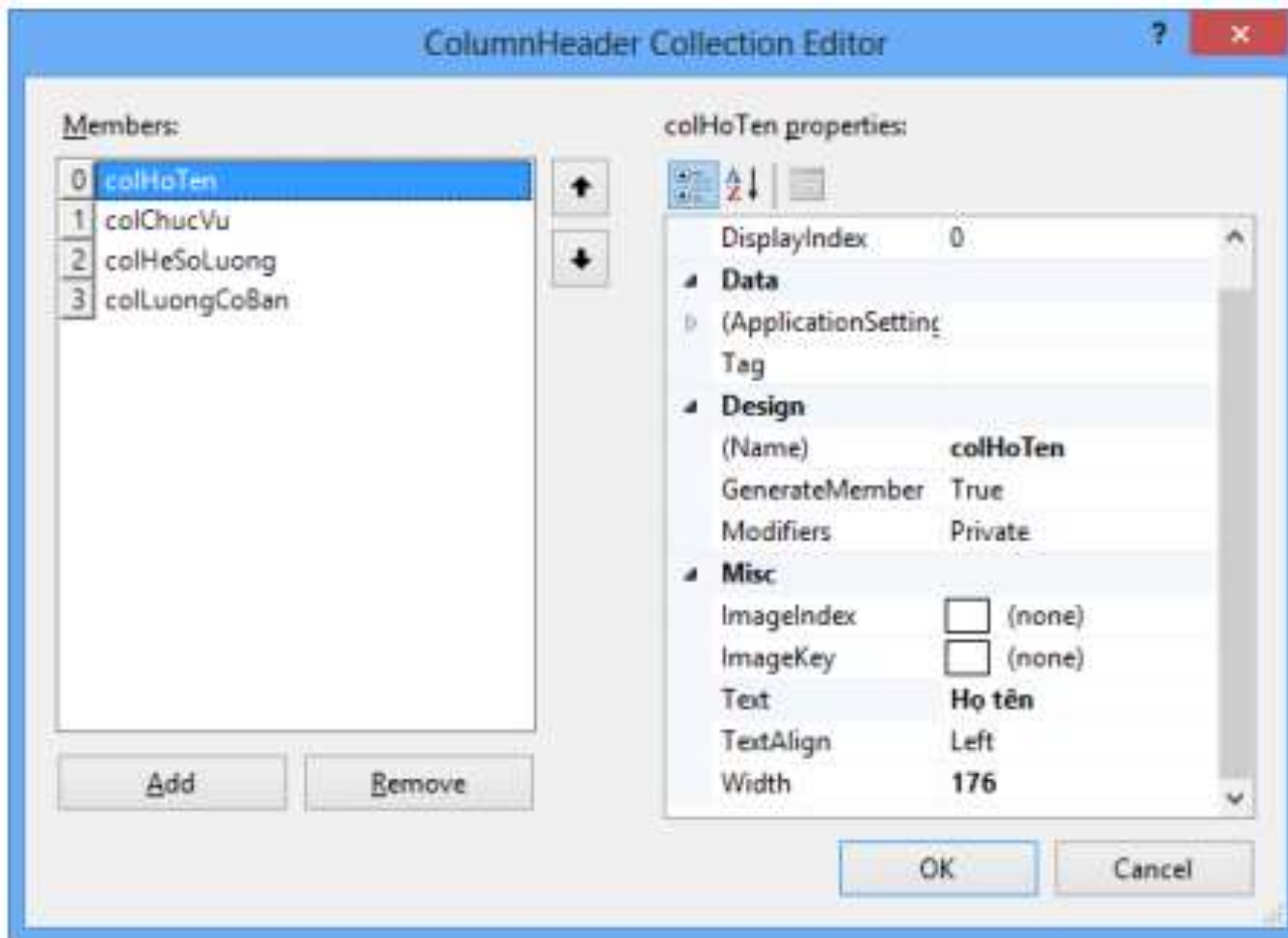
- Form1:
 - Thuộc tính *Text*: “TabControl” - label1:
 - Thuộc tính *Text*: “Quản lý nhân sự”
 - Thuộc tính *Size*: 14 - label2:
 - Thuộc tính *Text*: “Họ tên:” - label3:
 - Thuộc tính *Text*: “Lương cơ bản:”
- label4:
 - Thuộc tính *Text*: “Chức vụ:”
- label5:
 - Thuộc tính *Text*: “Hệ số lương:”
- textBox1:
 - Thuộc tính *Name*: txtHoTenNV
- textBox2:
 - Thuộc tính *Name*: txtLuongCBNV
- textBox3:
 - Thuộc tính *Name*: txtChucVuNV
- textBox4:
 - Thuộc tính *Name*: txtHeSoLuongNV - button1:
 - Thuộc tính *Name*: btnThoat
 - Thuộc tính *Text*: “Thoát”
- button2:
 - Thuộc tính *Name*: btnThemNV
 - Thuộc tính *Text*: “Thêm nhân viên”
- button3:
 - Thuộc tính *Name*: btnCapNhatNV
 - Thuộc tính *Text*: “Cập nhật” - button4:
 - Thuộc tính *Name*: btnXoaNV
 - Thuộc tính *Text*: “Xóa nhân viên”

4.6.5. Điều khiển TabControl

- **listView1:**

- ✓ Thuộc tính *Name*: listNhanVien
- ✓ Thuộc tính *View*: Detail
- ✓ Thuộc tính *FullRowSelect*: True
- ✓ Thuộc tính *MultiSelect*: False
- ✓ Thuộc tính *Columns*: Mở cửa sổ ColumnHeader Collection Editor, thêm 4 cột: colHoTen, colChucVu, colHeSoLuong, colLuongCoBan như hình.
 - Cột colHoTen: Thiết lập thuộc tính *Text* là “Họ tên”
 - Cột colChucVu: Thiết lập thuộc tính *Text* là “Chức vụ”
 - Cột colHeSoLuong: Thiết lập thuộc tính *Text* là “Hệ số lương”
 - Cột colLuongCoBan: Thiết lập thuộc tính *Text* là “Lương cơ bản”

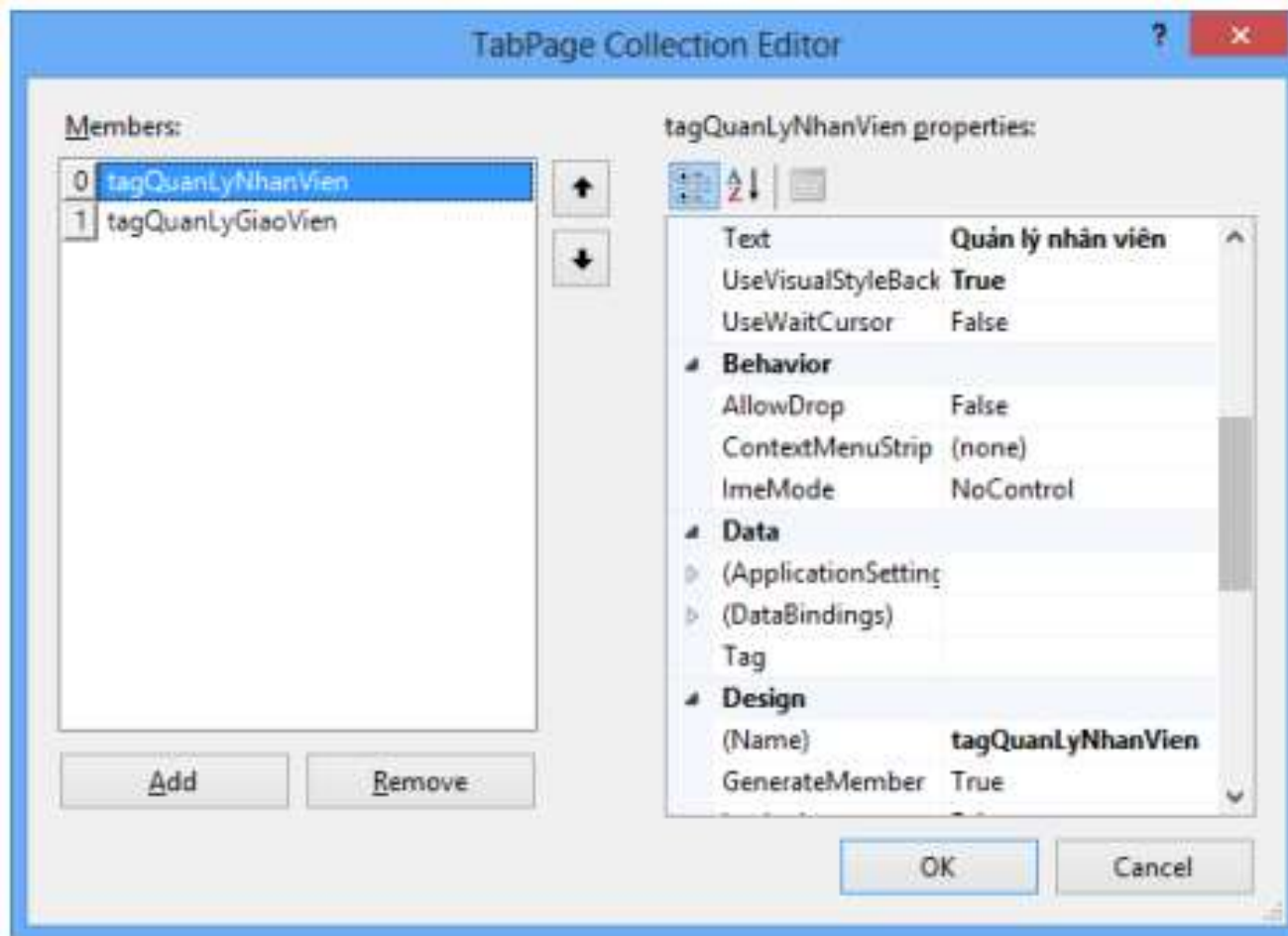
4.6.5. Điều khiển TabControl



4.6.5. Điều khiển TabControl

- tabControl1:
 - ✓ Thuộc tính Name: myTabControl
 - ✓ Thuộc tính *TabPage*: Mở cửa sổ TabPage Collection Editor, thêm 2 *TabPage*: Quản lý nhân viên và Quản lý giáo viên như hình.
 - ✓ Trên tabQuanLyNhanVien:
 - Thiết lập thuộc tính *Text*: “Quản lý nhân viên” Trên
 - ✓ tabQuanLyGiaoVien:
 - Thiết lập thuộc tính *Text*: “Quản lý giáo viên”

4.6.5. Điều khiển TabControl



4.6.5. Điều khiển TabControl

- Sự kiện *Click* của btnCapNhatNV

```
private void btnCapNhatNV_Click(object sender, EventArgs e)
{
    foreach (ListViewItem lvi in listNhanVien.SelectedItems)
    {
        lvi.SubItems[0].Text = txtTenNV.Text;
        lvi.SubItems[1].Text = txtChucVuNV.Text;
        lvi.SubItems[2].Text = txtHeSoLuongNV.Text;
        lvi.SubItems[3].Text = txtLuongCBNV.Text;
    }
}
```

- Sự kiện *Click* của btnThemNV:

```
private void btnThemNV_Click(object sender, EventArgs e)
{
    if (txtLuongCBNV.Text != "" && txtTenNV.Text != "" &&
        txtHeSoLuongNV.Text != "" && txtChucVuNV.Text != "")
    {
        ListViewItem LVItem = new ListViewItem(txtTenNV.Text);
        ListViewItem.ListViewSubItem LVSCV = new
        ListViewItem.ListViewSubItem(LVItem,
        txtChucVuNV.Text);
        ListViewItem.ListViewSubItem LVSHSL = new
        ListViewItem.ListViewSubItem(LVItem,
        txtHeSoLuongNV.Text);
        ListViewItem.ListViewSubItem LVSLCB = new
        ListViewItem.ListViewSubItem(LVItem,
        txtLuongCBNV.Text);
        LVItem.SubItems.Add(LVSCV);
        LVItem.SubItems.Add(LVSHSL);
        LVItem.SubItems.Add(LVSLCB);
        listNhanVien.Items.Add(LVItem);
        txtLuongCBNV.Text = "";
        txtTenNV.Text = "";
        txtHeSoLuongNV.Text = "";
        txtChucVuNV.Text = "";
    }
    else
        MessageBox.Show("Phải nhập đầy đủ thông tin nhân viên");
}
```

4.6.5. Điều khiển TabControl

- Sự kiện *Click* của btnXoaNV

```
private void btnXoaNV_Click(object sender, EventArgs e)
{
    if (listNhanVien.SelectedIndices.Count > 0)
    {
        listNhanVien.Items.RemoveAt(listNhanVien.FocusedItem.Index);
    }
    else
    {
        MessageBox.Show("Phải chọn nhân viên muốn ");
    }
}
```

4.6.5. Điều khiển TabControl

- Sự kiện *SelectedIndexChanged* của listNhanVien:

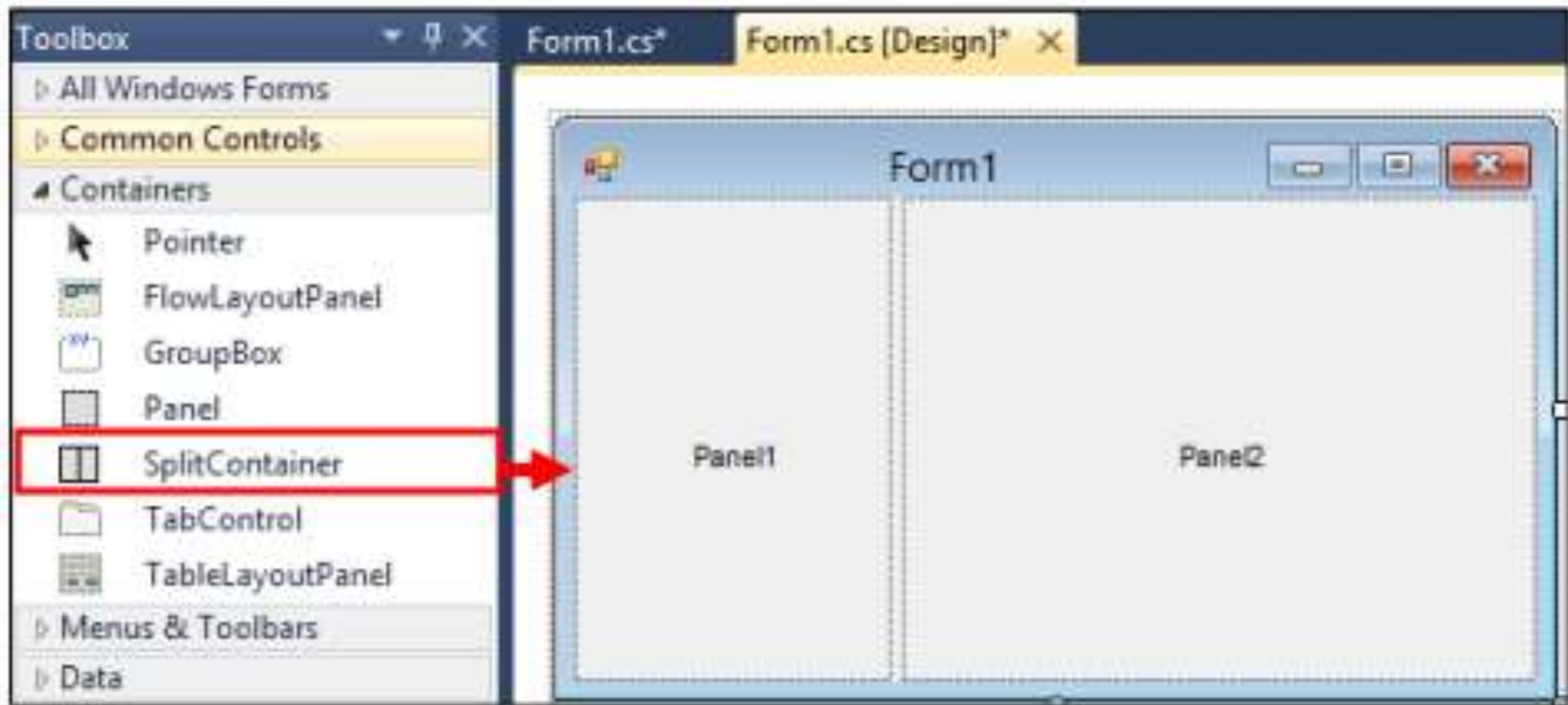
```
private void listNhanVien_SelectedIndexChanged(object sender, EventArgs e)
{
    foreach (ListViewItem lvi in listNhanVien.SelectedItems)
    {
        txtTenNV.Text = lvi.SubItems[0].Text;
        txtChucVuNV.Text = lvi.SubItems[1].Text;
        txtHeSoLuongNV.Text = lvi.SubItems[2].Text;
        txtLuongCBNV.Text = lvi.SubItems[3].Text;
    }
}
```

Tạo TagPage Quản lý giáo viên: Thiết kế và viết mã lệnh tương tự như TagPage Quản lý nhân viên.

4.6.6. Điều khiển SplitContainer

- Điều khiển *SplitContainer* giúp phân chia form thành hai phần. Cụ thể hơn *SplitContainer* được cấu tạo bởi hai điều khiển *Panel*, mỗi *Panel* trong điều khiển *SplitContainer* đều có chức năng như một điều khiển *Panel* thông thường.
- Khi thêm điều khiển *SplitContainer* từ cửa sổ Toolbox vào form thì mặc định *SplitContainer* có thuộc tính *Dock* mang giá trị *Fill*.
- Kích thước của *hai Panel* trong *SplitContainer* có thể thay đổi nhờ *Splitter*, *Splitter* là một vạch phân cách hai *Panel*

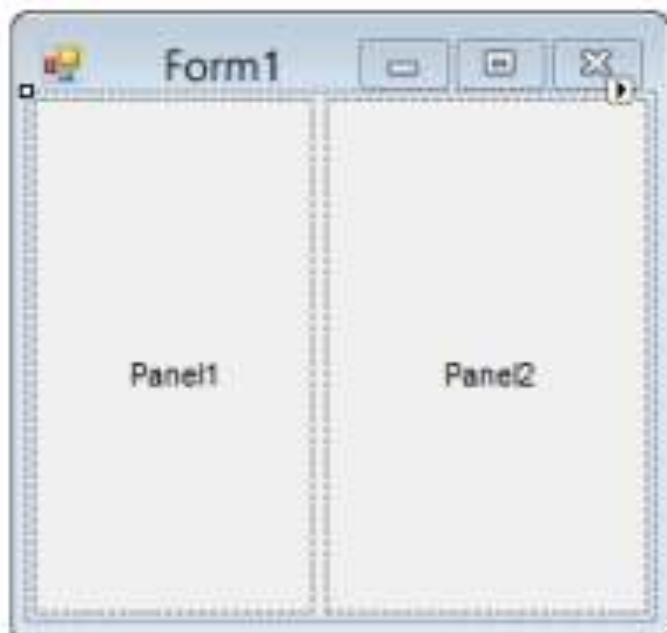
4.6.6. Điều khiển SplitContainer



4.6.6. Điều khiển SplitContainer

- Các *Panel* đều hỗ trợ thanh trượt (ScrollBar) khi thuộc tính *AutoScroll* được thiết lập là *True*.
- Tuy nhiên *Panel* trong *SplitContainer* không có thuộc tính *BorderStyle* để thiết lập đường viền, vì là điều khiển chứa trong *SplitContainer*, do đó thuộc tính *BorderStyle* được đặt ở điều khiển *SplitContainer*.
- Các giá trị của thuộc tính *BorderStyle* của *SplitContainer* cũng gồm 3 giá trị như *Panel*: *None*, *Fixed3D*, *FixedSingle*.
- Vạch phân cách *Splitter* có thể phân cách theo chiều dọc hoặc chiều ngang tùy thuộc vào thuộc tính *Orientation*.
- Thuộc tính *Orientation* mang hai giá trị để thiết lập *SplitContainer* là: *Vertical* và *Horizontal*

4.6.6. Điều khiển SplitContainer



*Thuộc tính Orientation là
Vertical*



*Thuộc tính Orientation là
Horizontal*

4.6.6. Điều khiển SplitContainer

- Nếu không muốn cho người dùng dịch chuyển vạch phân cách *Splitter* để thay đổi kích thước của hai *Panel*, lập trình viên có thể thiết lập thuộc tính *IsSplitterFixed* của *SplitContainer* là *True*
- Ngoài ra một điểm đặc biệt là có thể chỉ định không cho phép thay đổi kích thước của *Panel1* hoặc *Panel2* bằng cách kết hợp thuộc tính *FixedPanel* và thuộc tính *IsSplitterFixed*

4.6.6. Điều khiển SplitContainer

Thuộc tính	Mô tả
<i>IsSplitterFixed</i> = False	Thuộc tính <i>FixedPanel</i> không có hiệu lực. Người dùng có thể thay đổi kích thước của Panel1 và cả Panel2
<i>IsSplitterFixed</i> = True	Thiết lập thuộc tính FixedPanel: <ul style="list-style-type: none">- <i>FixedPanel</i> = None: Người dùng không thể sử dụng vạch phân cách <i>Splitter</i> để thay đổi kích thước của cả <i>Panel1</i> và <i>Panel2</i>. Nhưng kích thước cả hai Panel sẽ thay đổi khi <i>SplitContainer</i> có thuộc tính <i>Dock</i> là <i>Fill</i> và người dùng thay đổi kích thước form.- <i>FixedPanel</i> = Panel1: Người dùng không thể thay đổi kích thước Panel1 (Khi thay đổi kích thước form thì kích thước Panel2 thay đổi, kích thước Panel1 là không đổi).- <i>FixedPanel</i> = Panel2: Người dùng không thể thay đổi kích thước Panel2 (Khi thay đổi kích thước form thì kích thước Panel1 thay đổi, kích thước Panel2 là không đổi).

4.6.6. Điều khiển SplitContainer

- Một trong hai *Panel* của *SplitContainer* có thể ẩn đi bằng cách thiết lập thuộc tính *Panel1Collapsed* và *Panel2Collapsed* là *True*.
- Việc ẩn hai *Panel* chỉ có tác dụng với một *Panel*. Nghĩa là chỉ có thể thiết lập một trong hai thuộc tính *Panel1Collapsed* và *Panel2Collapsed* là *True*. Khi *Panel1Collapsed* là *True* thì *Panel2Collapsed* là *False* và ngược lại.

4.6.6. Điều khiển SplitContainer

- Các thuộc tính thường dùng của *SplitContainer*

Thuộc tính	Mô tả
<i>BorderStyle</i>	Thiết lập đường viền cho <i>SplitContainer</i> . Gồm 3 giá trị: <i>None</i> , <i>FixedSingle</i> , <i>Fixed3D</i> .
<i>FixedPanel</i>	Cố định kích thước của các <i>Panel</i> trong <i>SplitContainer</i> . Gồm 3 giá trị: <i>None</i> , <i>Panel1</i> , <i>Panel2</i> .
<i>IsSplitterFixed</i>	Mang hai giá trị True và False. Nếu là True, cố định vạch phân cách Splitter
<i>Orientation</i>	Xác định vạch phân cách Splitter sẽ phân cách theo chiều ngang hay dọc. Gồm 2 giá trị: <i>Vertical</i> , <i>Horizontal</i> .
<i>Panel1Collapsed</i>	Mang hai giá trị True hoặc False. Nếu là True sẽ ẩn <i>Panel1</i>

4.6.6. Điều khiển SplitContainer

Thuộc tính	Mô tả
<i>Panel1MinSize</i>	Lấy kích thước nhỏ nhất hoặc thiết lập kích thước nhỏ nhất cho Panel1
<i>Panel2Collapsed</i>	Mang hai giá trị True hoặc False. Nếu là True sẽ ẩn Panel2
<i>Panel2MinSize</i>	Lấy kích thước nhỏ nhất hoặc thiết lập kích thước nhỏ nhất cho Panel2
<i>SplitterDistance</i>	Trả về khoảng cách bằng pixel từ Splitter đến cạnh bên trái (nếu hai Panel nằm dọc) hay đến cạnh trên (nếu hai Panel nằm ngang)
<i>SplitterWidth</i>	thiết lập độ rộng của vạch phân cách Splitter

4.6.6. Điều khiển SplitContainer

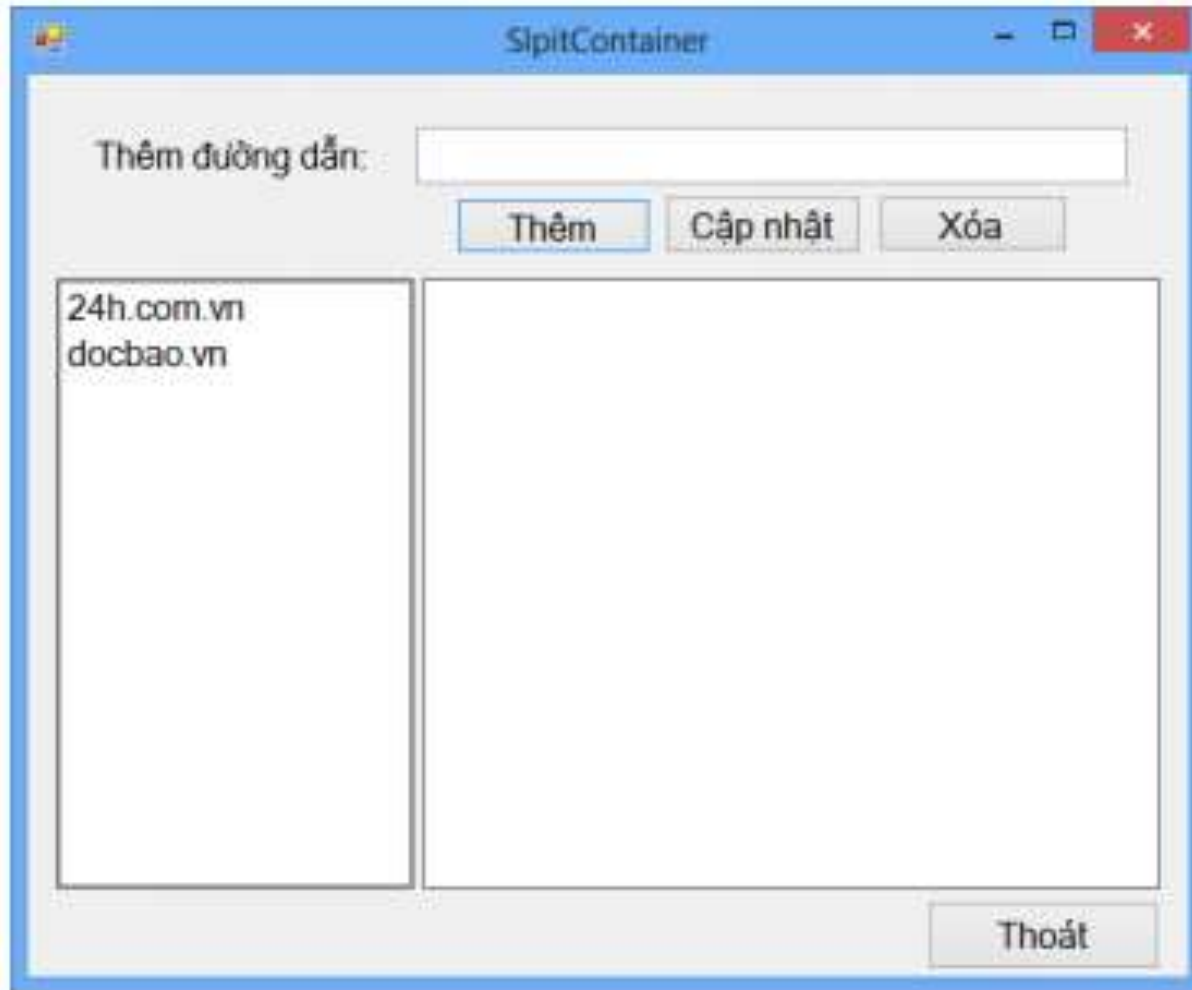
- Ví dụ:

Viết chương trình có giao diện như hình 6.27. Gồm: 1 điều khiển SplitContainer với Panel1 chứa 1 điều khiển ListView và Panel2 chứa 1 điều khiển WebBrowser.

Yêu cầu:

Người dùng nhập đường dẫn website vào TextBox và nhấn nút thêm. Đường dẫn website vừa nhập sẽ được đưa vào ListView. Người dùng có thể hiển thị bất cứ website nào trên WebBrowser bằng cách nhấp chuột vào đường dẫn chứa trong ListView

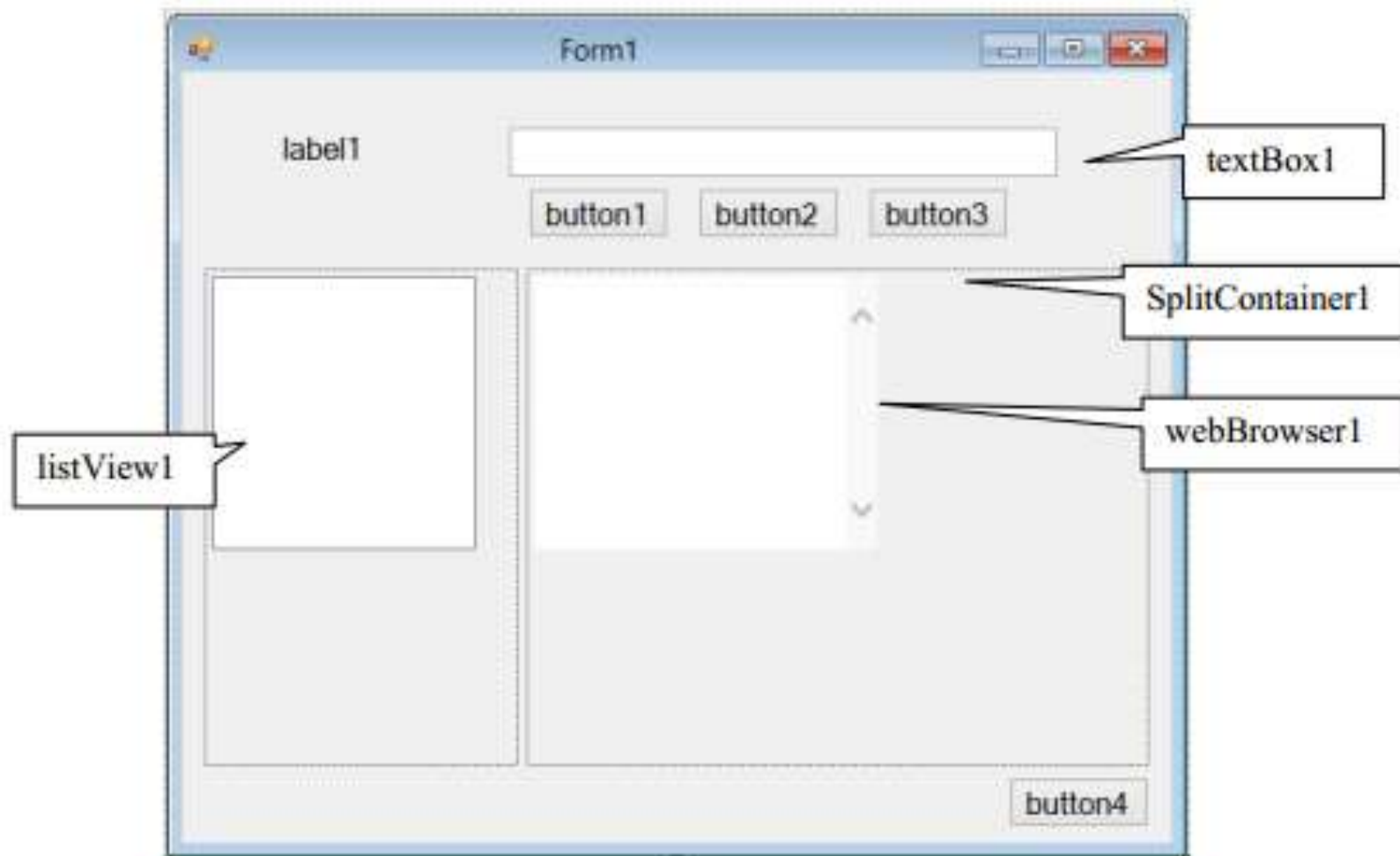
4.6.6. Điều khiển SplitContainer



4.6.6. Điều khiển SplitContainer

- Bước 1: Thiết kế giao diện ban đầu cho form. Thêm các điều khiển Label, TextBox, Button, SplitContainer và ListView, WebBrowser từ cửa sổ Toolbox vào form như hình

4.6.6. Điều khiển SplitContainer



4.6.6. Điều khiển SplitContainer

- Bước 2: Thiết lập giá trị thuộc tính cho các điều khiển trong cửa sổ Properties
 - ✓ Form1:
 - Thuộc tính Text: “SplitContainer”
 - ✓ label1:
 - Thuộc tính Text: “Thêm đường dẫn:”
 - ✓ textBox1:
 - Thuộc tính Name: txtLink
 - ✓ button1:
 - Thuộc tính Name: btnThem
 - Thuộc tính Text: “Thêm”
 - ✓ button2:
 - Thuộc tính Name: btnCapNhat
 - Thuộc tính Text: “Cập nhật”
 - ✓ button3:
 - Thuộc tính Name: btnXoa
 - Thuộc tính Text: “Xóa”

4.6.6. Điều khiển SplitContainer

✓ button4:

- Thuộc tính Name: btnThoat
- Thuộc tính Text: “Thoát”

✓ listView1:

- Thuộc tính Name: listLinkWebsite
- Thuộc tính Dock: Fill

✓ webBrowser1:

- Thuộc tính Name: myWebsite
- Thuộc tính Dock: Fill

✓ splitContainer1:

- Thuộc tính BorderStyle: FixedSingle
- Thuộc tính Dock: None
- Thuộc tính Orientation: Vertical
- Thuộc tính IsSplitterFixed: True

- Sự kiện *Load* của Form1

```
private void Form1_Load(object sender, EventArgs e)
{
    myWebsite.Navigate("www.google.com");
}
```

- Sự kiện *Click* của nút btnThem

```
private void btnThem_Click(object sender, EventArgs e)
{
    ListViewItem lvi = new ListViewItem(txtLink.Text);
    listLinkWebsite.Items.Add(lvi);
    txtLink.Text = "";
}
```

- Sự kiện *Click* của nút btnCapNhat

```
private void btnCapNhat_Click(object sender, EventArgs e)
{
    listLinkWebsite.FocusedItem.Text = txtLink.Text;
}
```


- Sự kiện **Click** của nút btnXoa

```
private void btnXoa_Click(object sender, EventArgs e)
{
    int i = listLinkWebsite.FocusedItem.Index;
    listLinkWebsite.Items.RemoveAt(i);
    txtLink.Text = "";
    myWebsite.Navigate("www.google.com");
}
```

- Sự kiện **MouseClick** của listLinkWebsite

```
private void listLinkWebsite_MouseClick(object sender,
MouseEventArgs e)
{
    txtLink.Text = listLinkWebsite.FocusedItem.Text;
    if (e.Button == MouseButton.Left)
        myWebsite.Navigate(listLinkWebsite.FocusedItem.Text);
}
```