

# LẬP TRÌNH GIAO DIỆN

Nguyễn Thị Mai Trang

Nguyễn Thị Mai Trang

1

1

## Chương 4

# Windows Form và các Control

2

## Mục tiêu

- Sử dụng đúng và hợp lý các loại giao diện ứng dụng
- Sử dụng thành thạo các thuộc tính của Form và control để xây dựng các ứng dụng đáp ứng yêu cầu ở mức độ từ cơ bản đến nâng cao
- Nắm bắt và xử lý các sự kiện trong Windows Form và các control một cách hợp lý
- Thao tác thành thạo khi chuyển form và truyền dữ liệu giữa các form
- Xây dựng và xử lý thành thạo ứng dụng dạng MDI

Nguyễn Thị Mai Trang

3

3

## Nội dung

- |                                |  |
|--------------------------------|--|
| 1. Form                        | 7. UserControl                                 |
| 2. Các controls                | 8. Thêm các controls lúc chương trình thực thi |
| 3. Các controls cơ bản         | 9. Menu  |
| 4. Các controls chứa           | 10. Các hộp thoại thông dụng                   |
| 5. Các controls dạng danh sách | 11. Ứng dụng SDI – MDI                         |
| 6. Các controls khác           |  |

Nguyễn Thị Mai Trang

4

4

## 4.1 Form

- Là cửa sổ chính của ứng dụng giao diện người dùng dạng đồ họa.
- Cung cấp giao diện tương tác với người sử dụng bằng thao tác trực quan.
- Trong ứng dụng Windows Forms, khi project được tạo, luôn có sẵn một form chính.
- Có thể bổ sung thêm nhiều form khác
- Khi chương trình thực thi, chỉ duy nhất một form được gọi.

## Form (tt)

- **Các thuộc tính của Form**
  - Name: Tên Form.
  - Text: Chuỗi hiển thị trên thanh tiêu đề.
  - ShowIcon: true/false – hiển thị/không hiển thị icon ở bên trái thanh tiêu đề.
  - ShowInTaskBar: true/false – hiển thị/không hiển thị biểu tượng của form trên thanh Taskbar khi form được thực thi.
  - Icon: tên tập tin \*.ico làm biểu tượng trên thanh tiêu đề của form.
  - BackColor: màu nền của form.
  - ForeColor: màu của các chuỗi trên các control của form.
  - StartPosition: vị trí hiển thị form.
  - Opacity: độ rõ của form, mặc định là 100%.

## Form (tt)

### • Các thuộc tính của Form (tt)

- WindowStates: trạng thái của form khi thực thi:
  - Minimized (thu nhỏ).
  - Maximized (phóng to).
  - Normal (trạng thái như thiết kế).
- isMdiContainer: được sử dụng trong ứng dụng MDI.
  - true: form được chọn là MDI form (form cha).
  - false: form bình thường.
- TopMost:
  - true: form nằm chồng lên trên các cửa sổ khác.
  - false: form bình thường.
- FormBorderStyle: kiểu đường viền của form.
- MainMenuStrip: control MenuStrip gắn trên form

Nguyễn Thị Mai Trang

7

7

## Form (tt)

### • Một số phương thức của Form

- Close (): đóng form.
- Hide (): ẩn form.
- Show (): Hiển thị form dạng modeless-dialog (khi form hiển thị, người sử dụng vẫn có thể thao tác được với các thành phần khác trong cùng một ứng dụng).
- ShowDialog (): Hiển thị form dạng modal-dialog (khi form hiển thị, người sử dụng không thể thao tác được với các thành phần khác trong cùng một ứng dụng cho đến khi đóng form).

Nguyễn Thị Mai Trang

8

8

## Form (tt)

### • Các sự kiện trên Form

- FormClosed: được gọi tự động khi form đã đóng.
- FormClosing: được gọi tự động khi form đang đóng.
- Click: được gọi tự động khi click chuột lên form.
- Activated: được gọi tự động khi form được kích hoạt bằng mã lệnh hay do tác động của người sử dụng
- Load: được gọi tự động khi form được nạp, dùng để khởi tạo giá trị các thành phần trong form.
- KeyPress, KeyDown, KeyUp: được gọi tự động khi một phím được nhấn trên form.
- Resize: được gọi tự động khi form bị thay đổi kích thước.

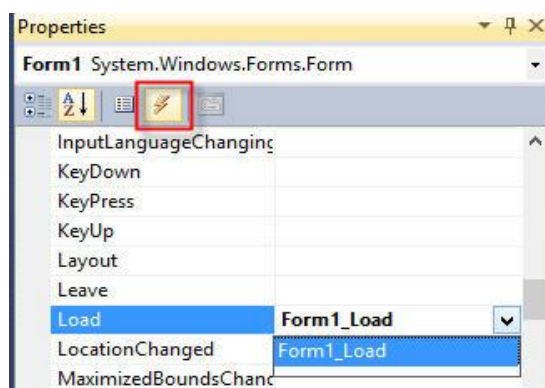
Nguyễn Thị Mai Trang

9

9

## Form (tt)

### • Cài đặt sự kiện trên Form: trong bảng properties, chọn tab Events

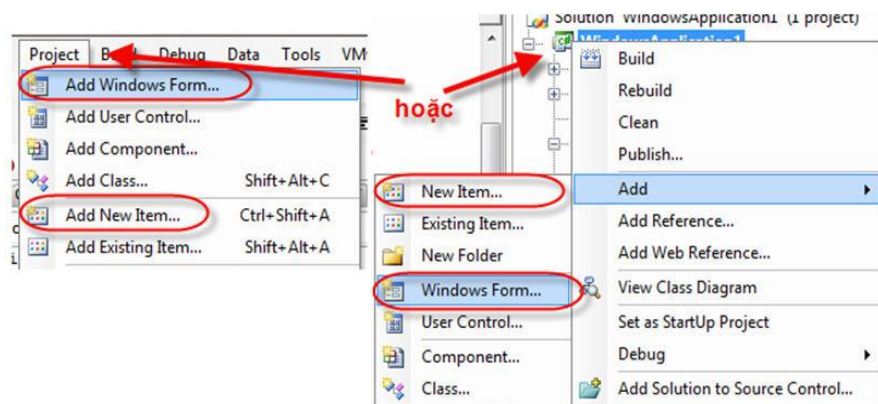


Nguyễn Thị Mai Trang

10

10

## Thêm một form vào project

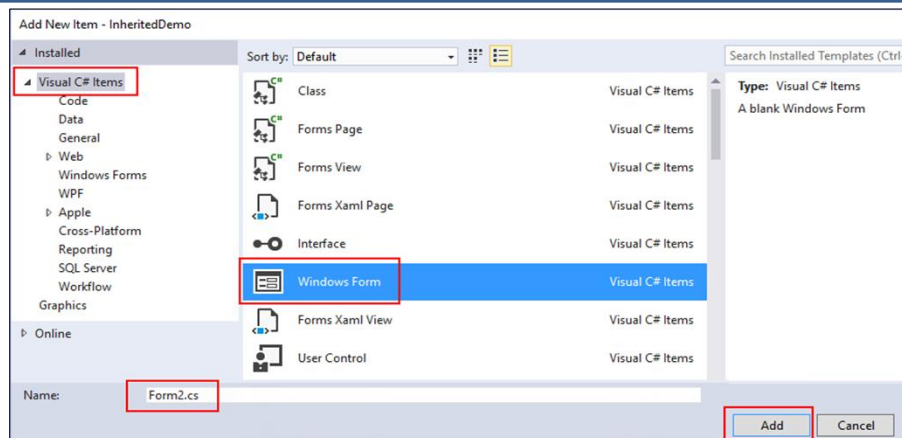


Nguyễn Thị Mai Trang

11

11

## Thêm một form vào project (tt)

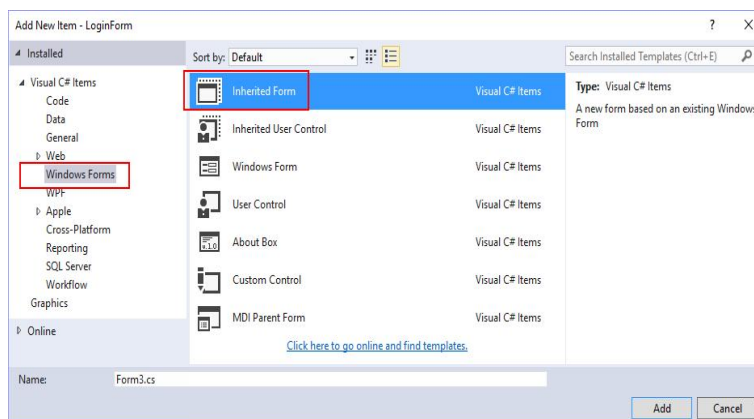


Nguyễn Thị Mai Trang

12

12

## Tạo Form kế thừa (Inherited Form)



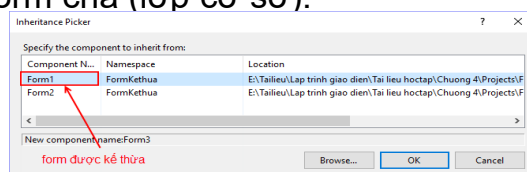
Nguyễn Thị Mai Trang

13

13

## Tạo Form kế thừa (tt)

- Chọn Form cha (lớp cơ sở):



- Tạo nhanh form kế thừa:

```
public partial class Form3 : Form1
{
    public Form3()
    {
        InitializeComponent();
    }
}
```

Nguyễn Thị Mai Trang

14

14

## Tạo Form lúc chương trình thực thi

- Tạo đối tượng thuộc class Form
- Xây dựng các thuộc tính cho đối tượng
- Gọi phương thức Show, hoặc ShowDialog

```
Form f = new Form();
f.Text = "New Form";
f.BackColor = Color.Red;
f.StartPosition = FormStartPosition.CenterParent;
f.ShowDialog();
```

## 4.2 Controls

- Controls là các thành phần mà ta có thể bổ sung lên form khi thiết kế giao diện cho chương trình.
- Ví dụ: các nút lệnh, các ô cho phép người sử dụng nhập liệu, các loại danh sách lựa chọn, các hộp kiểm, hình ảnh,...
- Các control được tổ chức thành các nhóm nằm trên cửa sổ Toolbox, cho phép người sử dụng kéo thả vào form một cách trực quan.



## Controls (tt)

- Thêm các controls vào Form:
  - Hộp công cụ (Toolbox): cung cấp danh sách các Component liệt kê theo nhóm, cho phép thiết kế giao tiếp với người dùng.
  - Hiện Toolbox:
    - View, Toolbox
    - Chọn biểu tượng trên thanh công cụ
    - Ctrl+W và X



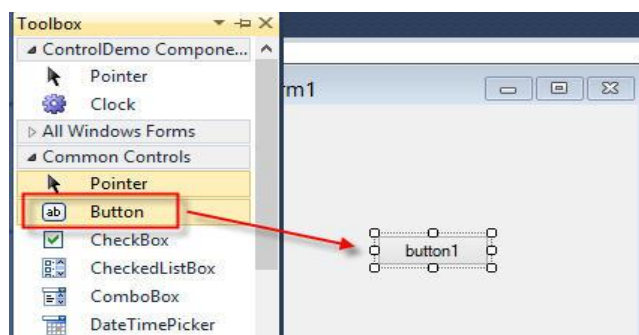
Nguyễn Thị Mai Trang

17

17

## Controls (tt)

- Thêm các controls vào Form:
  - Kéo thả control từ hộp Toolbox vào Form



Nguyễn Thị Mai Trang

18

18

## Controls (tt)

### • Thuộc tính chung của các control:

- BackColor: Màu nền
- ForeColor: Màu chữ trên control
- Text: Chuỗi hiển thị trên control
- Visible: Ẩn hay hiển thị control
- Name: Tên của control, dùng để truy xuất các thuộc tính của control
- Locked: Khoá không cho di chuyển trên Form
- Enabled: Vô hiệu hoá hay cho phép sử dụng

## Controls (tt)

### • Sự kiện chung của các control:

- Click: click chuột lên control.
- MouseMove: di chuyển chuột trên control.
- MouseDown: nhấn chuột trên control.
- MouseUp: nhả chuột sau khi nhấn chuột trên control.
- Move: di chuyển control bằng mã lệnh hay sử dụng chuột.
- SizeChanged: kích thước control được thay đổi bằng mã lệnh hay do người sử dụng.
- Paint: xảy ra khi control được vẽ lại.

## Controls (tt)

- **Xếp thứ tự các control:**
  - View, Tab Order
  - Click chuột lần lượt trên các control để thay đổi số thứ tự (là thứ tự các control trên Form)
- **Sắp xếp các control:**
  - Sử dụng menu Format
  - Sử dụng thanh công cụ
- **Thay đổi thuộc tính các control:**
  - Click chuột phải trên control, chọn Properties
  - Chọn các thuộc tính cần thay đổi trên cửa sổ Properties

21

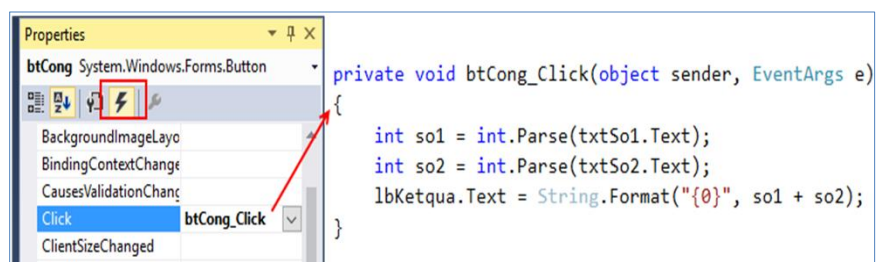
## Controls (tt)

- **Định vị các control:**
  - Sử dụng code: thiết lập giá trị các thuộc tính Size, Location, Top, Left, Width, Height.
  - Sử dụng thuộc tính **Anchor**
  - Sử dụng thuộc tính **Dock**

22

## Controls (tt)

- Xử lý các sự kiện của control:
  - Click chuột phải trên control, chọn Properties
  - Double click lên sự kiện trên tab Event trên cửa sổ Properties



Nguyễn Thị Mai Trang

23

23

## 4.3 Các control cơ bản

- Label
- TextBox
- Button
- CheckBox
- RadioButton
- PictureBox
- NumericUpDown
- Tooltip
- VScroll, HScroll...

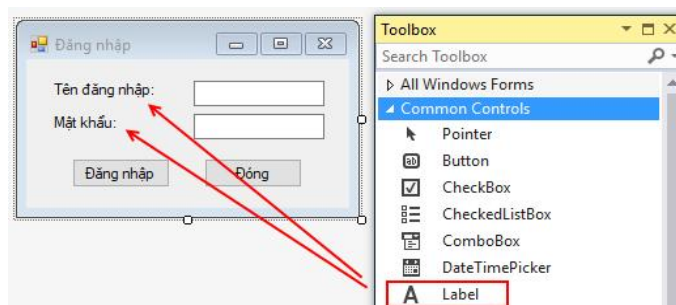
Nguyễn Thị Mai Trang

24

24

## Label

- Trình bày văn bản dạng “tĩnh”, thường được dùng để chú thích cho các control khác hoặc gợi ý cho người sử dụng.



Nguyễn Thị Mai Trang

25

25

## Label (tt)

- Các thuộc tính thông dụng:
  - **BorderStyle**: Kiểu đường viền của label.
  - **TextAlign**: Canh chỉnh văn bản trong label.
  - **AutoSize**:
    - true/false - cho / không cho phép label tự động thay đổi kích thước theo độ dài của chuỗi chứa bên trong nó.

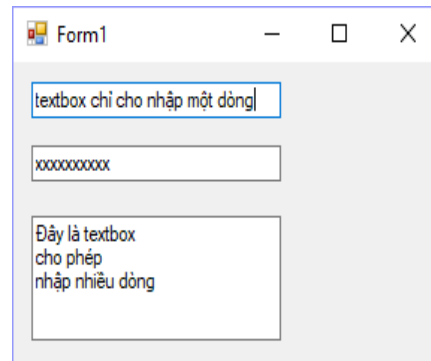
Nguyễn Thị Mai Trang

26

26

## TextBox

- Control được dùng để nhập dữ liệu
- Có ba dạng:
  - nhập một dòng văn bản
  - nhập nhiều dòng
  - nhập dạng mật khẩu



Nguyễn Thị Mai Trang

27

27

## TextBox (tt)

- **Một số thuộc tính:**
  - BorderStyle: kiểu đường viền của textbox.
  - CharacterCasing: định dạng kiểu chữ hoa (Upper), chữ thường (Lower) hay mặc định (Normal).
  - Maxlength: số ký tự cho phép nhập.
  - MultiLine: true/false - cho/không cho phép nhập nhiều dòng.
  - PasswordChar: ký tự thay thế ký tự nhập.
  - ReadOnly: true/false - khóa/cho phép nhập văn bản.
  - ScrollBars: thiết lập thanh cuộn khi MultiLine = true
    - Vertical: thanh cuộn dọc.
    - Horizontal: thanh cuộn ngang,
    - Both: cả 2 thanh cuộn,
    - None: không có thanh cuộn.
  - WordWrap: true/false - cho/không cho phép văn bản tự động xuống dòng khi chuỗi nhập dài hơn kích thước của điều khiển.

Nguyễn Thị Mai Trang

28

28

## TextBox (tt)

- Một số sự kiện trên TextBox:
  - **MouseClick**: xảy ra khi click vào TextBox.
  - **MouseDoubleClick**: xảy ra khi double click vào TextBox.
  - **TextChanged**: sự kiện mặc định, xảy ra khi chuỗi trên TextBox bị thay đổi.

Nguyễn Thị Mai Trang

29

29

## Button

- Nút lệnh, khi click vào sẽ thực thi một tác vụ nào đó.
- Hiện thị chuỗi hoặc hình ảnh



- Double click lên button để tạo sự kiện đáp ứng khi người dùng click chuột trên button.

```
private void button1_Click(object sender, EventArgs e)
{
    //viết code xử lý tại đây
}
```

Nguyễn Thị Mai Trang

30

30

## CheckBox

- Cho phép người sử dụng tại cùng một thời điểm có thể chọn nhiều lựa chọn.
- Các thuộc tính thông dụng:
  - **Checked**: true/false (được chọn/ không chọn)
  - **CheckState**: trạng thái được chọn của checkbox
    - Checked.
    - Unchecked.
    - Indeterminate.
- Sự kiện mặc định: **CheckedChanged**, xảy ra khi người sử dụng thay đổi lựa chọn trên checkbox.
- Ví dụ: xem tài liệu học tập Lập trình giao diện

Nguyễn Thị Mai Trang

31

31

## RadioButton

- Tương tự CheckBox, nhưng chỉ cho phép người sử dụng chọn một trong các lựa chọn.
- Sự kiện mặc định và cách xử lý tương tự như checkbox.



Nguyễn Thị Mai Trang

32

32



## PictureBox

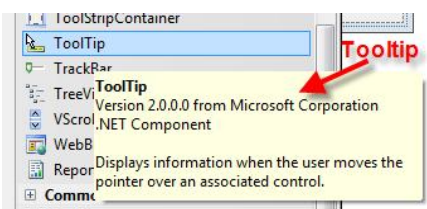
- Hiện thị hình ảnh trên giao diện
- Các dạng hình ảnh: BMP, JPEG, GIF, PNG, metafile, icon,...
- Một số thuộc tính cơ bản sau:
  - **Image**: đối tượng Image hiển thị hình.
  - **SizeMode**:
    - AutoSize.
    - CenterImage.
    - Normal.
    - StretchImage
    - Zoom:

## PictureBox (tt)

- Thiết lập thuộc tính Image cho PictureBox :
  - Bằng thiết kế trực quan
  - Bằng code
  - Ví dụ: xem tài liệu học tập Lập trình giao diện

## Tooltip

- Dùng để hiển thị chú thích cạnh các control khi đưa trỏ chuột đến control
- Một số thuộc tính:
  - TooltipTitle: chuỗi tiêu đề Tooltip
  - TooltipIcon: biểu tượng hiển thị kèm theo chuỗi



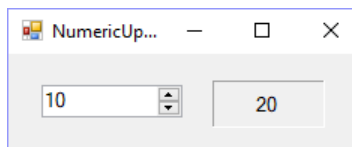
Nguyễn Thị Mai Trang

35

35

## NumericUpDown

- Control hiển thị số nguyên cho phép người sử dụng tăng, giảm giá trị khi chương trình thực thi.
- Một số thuộc tính cơ bản:
  - Value: giá trị đang hiển thị trên control.
  - Increment: giá trị mỗi lần tăng/giảm.
  - Minimum: giá trị nhỏ nhất trên control.
  - Maximum: giá trị lớn nhất trên control.
- Sự kiện mặc định: ValueChanged



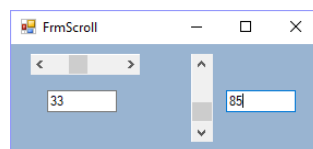
Nguyễn Thị Mai Trang

36

36

## VScrollBar, HScrollBar

- HScrollBar: thanh cuộn ngang
- VScrollBar: thanh cuộn dọc.
- Cho phép người sử dụng thay đổi giá trị khi chương trình thực thi.
- Một số thuộc tính cơ bản:
  - Value: giá trị hiện hành trên control.
  - Minimum: giá trị nhỏ nhất của control.
  - Maximum: giá trị lớn nhất của control.
- Sự kiện mặc định: Scroll.



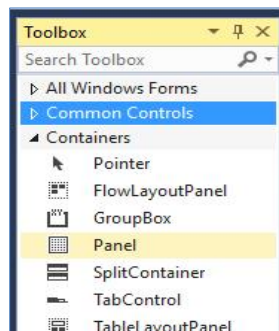
Nguyễn Thị Mai Trang

37

37

## 4.4 Các control chứa

- Còn gọi là Container control, là các loại control có thể chứa các control khác
- Được sử dụng khi cần xử lý chung một nhóm các control. Nhóm control này bao gồm:
  - GroupBox.
  - Panel, FlowLayoutPanel, TableLayoutPanel
  - TabControl.
  - SplitContainer.



Nguyễn Thị Mai Trang

38

38

## Các control chứa (tt)

- **GroupBox:**

- Hiển thị một khung bao quanh một nhóm control
- Có thể hiển thị một tiêu đề



- **Panel:**

- Không có tiêu đề
- Có thanh cuộn (scrollbar)
- Chứa nhiều control hơn GroupBox



Nguyễn Thị Mai Trang

39

39

## Các control chứa (tt)

- **TabControl:**

- Cho phép thể hiện nhiều trang trên cùng một form
- Các control có cùng nhóm chức năng thường được sắp xếp trong cùng một trang.
- Mỗi trang trong TabControl là một TabPage có chứa tiêu đề trang.
- Để chuyển qua lại giữa các trang, ta có thể click vào các tiêu đề trang.
- Để thêm, xóa và hiệu chỉnh một trang trong TabControl → truy cập thuộc tính TabPages và thao tác trong cửa sổ tabPage Collection Editor.

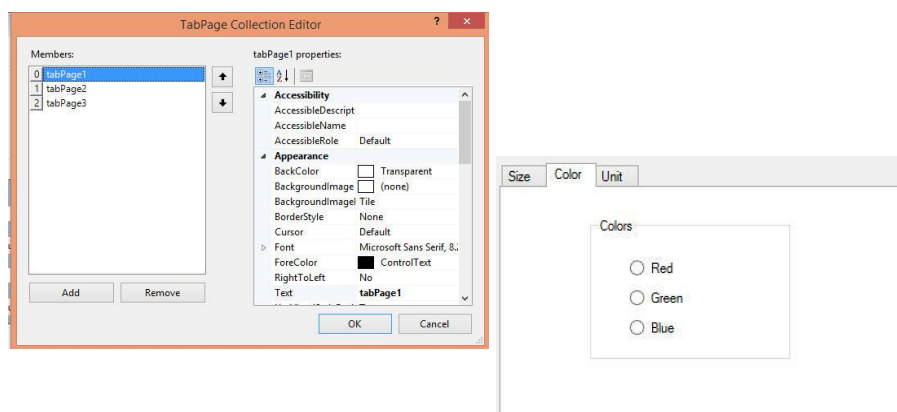
Nguyễn Thị Mai Trang

40

40

## Các control chứa (tt)

### – TabPage Collection Editor



Nguyễn Thị Mai Trang

41

41

## 4.5. Các control dạng danh sách

- ListBox
- ComboBox
- ImageList
- ListView
- TreeView

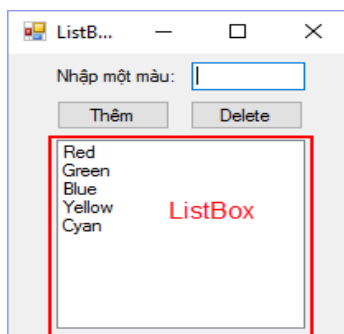
Nguyễn Thị Mai Trang

42

42

## ListBox

- Control hiển thị danh sách các phần tử là các chuỗi văn bản
- Cho phép chọn một hoặc nhiều phần tử.



Nguyễn Thị Mai Trang

43

43

## ListBox (tt)

- Một số thuộc tính cơ bản:
  - Items: danh sách các phần tử trong ListBox
  - SelectedIndex: vị trí phần tử được chọn.
  - SelectedItem: phần tử được chọn.
  - SelectedItems: danh sách các phần tử được chọn.
  - SelectionMode
    - None: không cho phép chọn.
    - One: chỉ cho phép chọn một phần tử.
    - MultiSimple: chọn nhiều phần tử, không giữ phím Ctrl, Shift.
    - MultiExtended: chọn nhiều phần tử, giữ phím Ctrl, Shift.
  - Sorted: cho phép xếp thứ tự tự động.

Nguyễn Thị Mai Trang

44

44

## ListBox (tt)

- Một số phương thức:
  - ClearSelected: bỏ chọn tất cả.
  - FindString: Tìm một chuỗi trong ListBox bắt đầu bởi một chuỗi cần tìm.
  - FindStringExact: như FindString, nhưng tìm chính xác.
  - GetSelected: trả về phần tử được chọn.
  - SetSelected: chọn một phần tử.

## ListBox (tt)

- Thêm phần tử vào ListBox: sử dụng các phương thức Add, Insert:
  - list\_name.Items.Add (text)
  - list\_name.Items.AddRange (array)
  - list\_name.Items.Insert (index, text)
- Xóa phần tử trong ListBox: sử dụng các phương thức Remove và RemoveAt:
  - list\_name.Items.RemoveAt (index)
  - list\_name.Items.Remove (delstring)

## ComboBox

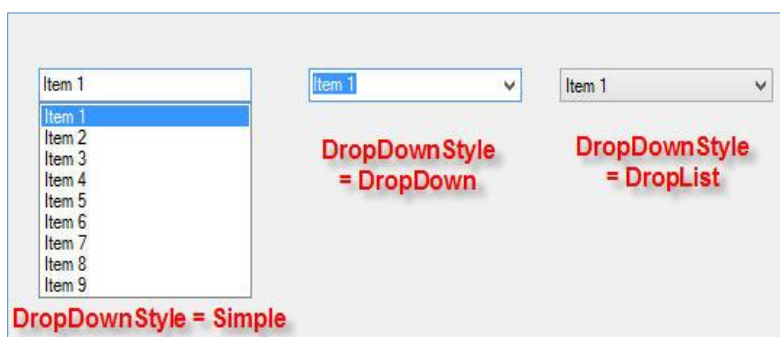
- Tương tự như ListBox, nhưng khác ở dạng thức trình bày.
- Một số thuộc tính riêng:
  - MaxDropDownItems: số phần tử được nhìn thấy khi xổ xuống.
  - DropDownStyle:
    - Simple: hiển thị danh sách xổ xuống, là sự kết hợp giữa textbox và listbox.
    - DropDown: cho phép chọn, đồng thời hiển thị textbox cho phép nhập vào một chuỗi.
    - DropDownList: chỉ cho phép chọn, không cho phép nhập chuỗi.

Nguyễn Thị Mai Trang

47

47

## ComboBox (tt)



Nguyễn Thị Mai Trang

48

48



## ImageList

- Control chứa danh sách các đối tượng Image làm nguồn cho các control khác như ListView, TreeView.
- Các thuộc tính cơ bản:
  - Images: danh sách các ảnh trong ImageList.
  - ColorDepth: độ sâu của màu.
  - ImageSize: kích thước ảnh.
  - TransparentColor: màu trong suốt.

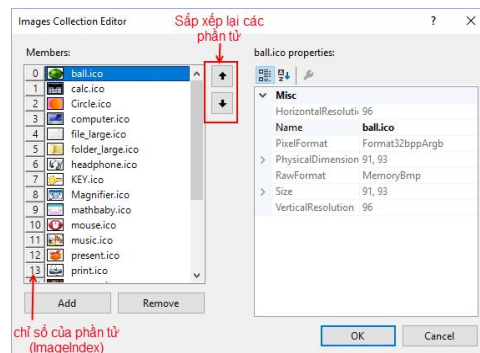
Nguyễn Thị Mai Trang

49

49

## ImageList (tt)

- Thêm ảnh vào ImageList bằng cửa sổ Images Collection Editor:



Nguyễn Thị Mai Trang

50

50

## Điều khiển ImageList (tt)

- Các bước sử dụng ImageList
  - Thêm control ImageList lên Form.
  - Thiết lập kích thước của các ảnh: ImageSize.
  - Bổ sung các ảnh vào ImageList: Images.
  - Sử dụng ImageList cho các control (như ListView, TreeView)
    - Khai báo nguồn image là image list vừa tạo cho control, thường là thuộc tính ImageList.
    - Thiết lập các item/node với ImageIndex tương ứng.
  - Việc thiết lập có thể ở màn hình design view hoặc code view

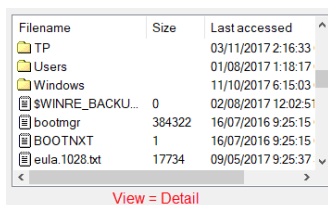
Nguyễn Thị Mai Trang

51

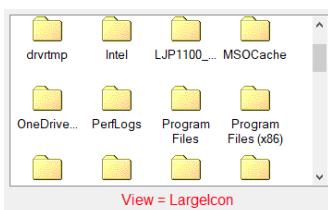
51

## ListView

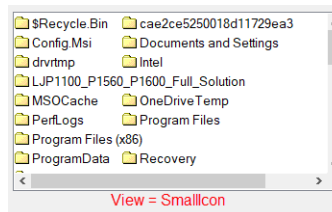
- Trình bày các phần tử dạng danh sách với nhiều dạng



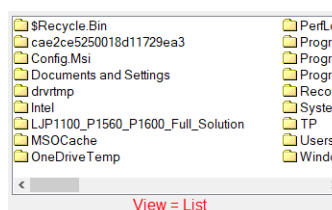
View = Detail



View = LargeIcon



View = SmallIcon



View = List

Nguyễn Thị Mai Trang

52

52

## ListView (tt)

- Một số thuộc tính cơ bản:
  - Columns: danh sách các cột trong ListView.
  - Items: danh sách các phần tử (hàng) trong ListView.
  - LargeImageList: ImageList, chứa các biểu tượng hiển thị trong ListView ở chế độ View=LargeIcon.
  - SmallImageList: ImageList, chứa các biểu tượng hiển thị trong ListView ở chế độ View=SmallIcon.
  - View: chế độ hiển thị của ListView.
  - SelectedItems: danh sách các phần tử được chọn trong ListView.

## ListView (tt)

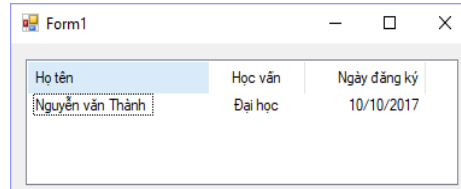
- Thiết kế ListView trong môi trường trực quan: sử dụng cửa sổ ListView Tasks.
- Thêm cột vào ListView bằng code:
 

```
listView1.Columns.Add ("ColumnName", Width, Alignment);
```

  - ColumnName: chuỗi trên tiêu đề cột.
  - Width: độ rộng cột, mặc định bằng với kích thước chuỗi.
  - Alignment: canh lề chuỗi trên tiêu đề cột:
    - HorizontalAlignment.Left (mặc định)
    - HorizontalAlignment.Right
    - HorizontalAlignment.Center

## ListView (tt)

- Thêm các phần tử vào ListView bằng code:
  - Mỗi phần tử là một đối tượng ListViewItem.
  - Tạo đối tượng ListViewItem, đó là phần tử có chuỗi hiển thị ở cột đầu tiên
  - Các chuỗi hiển thị ở các cột kế tiếp lần lượt là các đối tượng SubItem.
- Ví dụ: xem tài liệu học tập Lập trình giao diện



Họ tên	Học vấn	Ngày đăng ký
Nguyễn văn Thành	Đại học	10/10/2017

Nguyễn Thị Mai Trang

55

55

## ListView (tt)

- Xóa một phần tử trong ListView:
  - Xóa phần tử ListViewItem:
 

```
listView1.Items.Remove(item);
```
  - Xóa phần tử theo vị trí:
 

```
listView1.Items.RemoveAt(index);
```
  - Trong đó:
    - item là một đối tượng ListViewItem.
    - index là vị trí phần tử trong ListView.
- Thay đổi chế độ hiển thị của ListView:
 

```
listView1.View = View.Details;
```

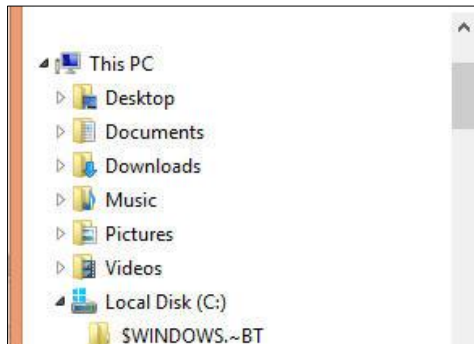
Nguyễn Thị Mai Trang

56

56

## TreeView

- Trình bày danh sách phần tử phân cấp theo từng node (tương tự Windows Explorer)



Nguyễn Thị Mai Trang

57

57

## TreeView (tt)

- Một số thuộc tính cơ bản:
  - Nodes: danh sách các phần tử trong TreeView.
  - ImageList: ImageList, chứa các biểu tượng hiển thị trong TreeView.
    - TreeView chỉ sử dụng duy nhất một đối tượng ImageList.
  - SelectedNode: node đang được chọn.
  - TopNode: node cao nhất trong TreeView.

Nguyễn Thị Mai Trang

58

58

## TreeView (tt)

- Thiết kế TreeView trong môi trường trực quan: sử dụng cửa sổ TreeView Tasks
- **Thêm node vào TreeView:** mỗi phần tử trong TreeView là một **TreeNode**
  - Thêm từng node vào TreeView:

```
treeView.Nodes.Add(treenode)
treeView.Nodes [level].Nodes.Add(treenode)
treeView.Nodes [level].Nodes[level1].Nodes.Add(treenode)
```
  - Tạo đối tượng TreeNode → thiết lập các thuộc tính cho đối tượng → thêm node mới tạo vào TreeView:

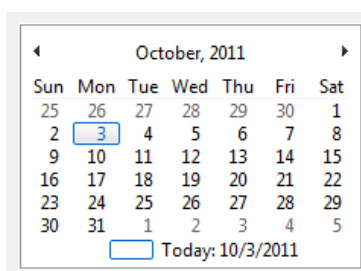
```
TreeNode newnode= new TreeNode ("node name");
newnode.ImageIndex = 0;
treeView1.Nodes.Add(newnode);
```

## 4.6 Các control khác

- Month Calendar
- DateTimePicker
- LinkLabel
- Timer
- RichTextBox

## MonthCalendar

- Thuộc tính
  - SelectionStart: ngày bắt đầu danh sách chọn
  - SelectionEnd: ngày cuối danh sách chọn
  - TodayDate: ngày hiện tại (trên máy tính)
- Biến cố mặc định: DateChanged



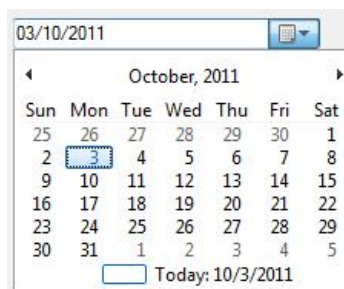
Nguyễn Thị Mai Trang

61

61

## DateTimePicker

- Kết hợp của hai control ComboBox + MonthCalendar.
- Không chiếm nhiều diện tích trên giao diện và tính năng sử dụng linh hoạt hơn MonthCalendar.
- Các thuộc tính:
  - Format: định dạng hiển thị
    - long, short, time, custom
  - CustomFormat:
    - xem thêm MSDN Online
  - MaxDate: giá trị ngày lớn nhất
  - MinDate: giá trị ngày nhỏ nhất
  - Value: giá trị ngày hiện tại đang chọn
- Sự kiện mặc định: ValueChanged



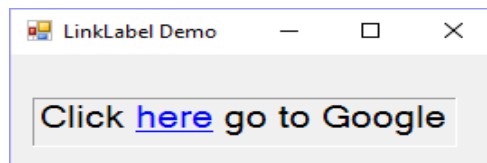
Nguyễn Thị Mai Trang

62

62

## LinkLabel

- Chứa liên kết đến một URL, text file,...



Nguyễn Thị Mai Trang

63

63

## Timer

- Là loại điều khiển cho phép thực thi một tác vụ sau một khoảng thời gian
- Các thuộc tính của Timer:
  - Name: định danh Timer
  - Enabled: true/false, start hoặc stop timer
  - Interval: khoảng thời gian kích hoạt sự kiện tick được tính bằng mili giây
- Sự kiện **Tick**: sau khoảng thời gian được thiết lập ở thuộc tính Interval, một sự kiện được gọi đến cửa sổ, cho phép xử lý

Nguyễn Thị Mai Trang

64

64



## RichTextBox

- Tương tự như TextBox, nhưng cung cấp nhiều khả năng định dạng hơn
- Cho phép nhập văn bản, hình ảnh hoặc tải nội dung từ một file .txt, .rtf, .docx,...
- Một số thuộc tính cơ bản:
  - SelectedText: chuỗi văn bản được chọn trên RichTextBox.
  - SelectionFont: font chữ áp dụng cho phần văn bản được chọn.
  - SelectionColor: màu chữ áp dụng cho phần văn bản được chọn.
  - CanFocus: true/false - RichTextBox có/không nhận focus.
  - CanPaste: true/false - RichTextBox có/không thể paste.
  - CanSelect: true/false - RichTextBox có/không cho phép chọn văn bản..
  - CanUndo: true/false - RichTextBox có/không thể undo.

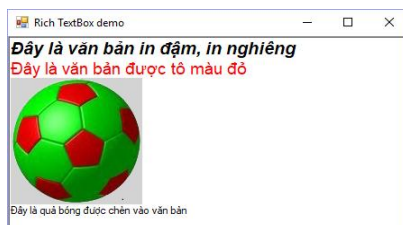
Nguyễn Thị Mai Trang

65

65

## RichTextBox (tt)

- Một số phương thức cơ bản :
  - Copy, Cut: sao chép, cắt dữ liệu được chọn trong RichTextBox lưu vào Clipboard.
  - Paste: dán dữ liệu từ Clipboard vào RichTextBox.
  - SaveFile: lưu dữ liệu trong RichTextBox ra file.
- Ví dụ: xem tài liệu học tập Lập trình giao diện



Nguyễn Thị Mai Trang

66

66

## 4.7 User control

- Là control do người sử dụng tự định nghĩa.
- Được sử dụng như các control thông thường khác bằng cách kéo thả từ cửa sổ Toolbox.
- UserControl được tạo ra còn có thể được sử dụng trong các ứng dụng khác.
- Tạo UserControl dùng trong một ứng dụng:
  - **Project → Add User Control**
  - UserControl được tạo ra giống như một form nhưng không có tiêu đề.
  - Thiết kế giao diện và thao tác trên UserControl như với một form thông thường.

## User control (tt)

- Tạo User control sử dụng trong ứng dụng khác
  - Tạo ứng dụng loại Class Library project.
  - Sau khi biên dịch thành công, ứng dụng tạo ra một file có phần mở rộng là dll.
  - Trong ứng dụng khác, click chuột phải trên cửa sổ Toolbox, chọn Choose item..., chọn file dll nói trên, click nút OK để hoàn tất.
  - UserControl sẽ hiển thị trên cửa sổ Toolbox như các control khác.

## 4.8 Thêm các control lúc chương trình thực thi

- Khai báo và tạo đối tượng control muốn thêm vào form.
- Thiết lập các thuộc tính cho đối tượng:
  - vị trí: Location, X, Y
  - kích thước: Size, Width, Height
  - ...
- Khai báo sự kiện cho control nếu cần.
- Thêm đối tượng control vào danh sách Controls của Form hoặc Control chứa (Panel, GroupBox, TabControl...).

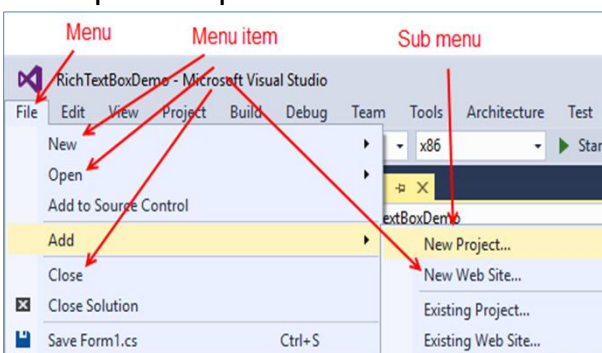
Nguyễn Thị Mai Trang

69

69

## 4.9 Menu

- Control cho phép tổ chức các chức năng xử lý của ứng dụng theo nhóm.
- Menu được tổ chức phân cấp
  - Menu
  - Sub menu
  - Menu item



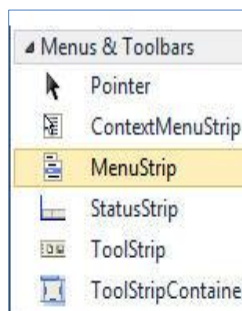
Nguyễn Thị Mai Trang

70

70

## Menu (tt)

- Tạo menu: sử dụng điều khiển MenuStrip
- Một số thuộc tính của Menu
  - Text: chuỗi hiển thị
  - Shortcut Keys: phím nóng kết hợp với menu
  - Image: hình ảnh hiển thị trên menu
  - AutoTooltip,...
- Các loại menu item
  - MenuItem
  - ComboBox
  - TextBox
  - Separator
- Sự kiện mặc định: Click



Nguyễn Thị Mai Trang

71

71

## 4.10 Các hộp thoại thông dụng của Windows

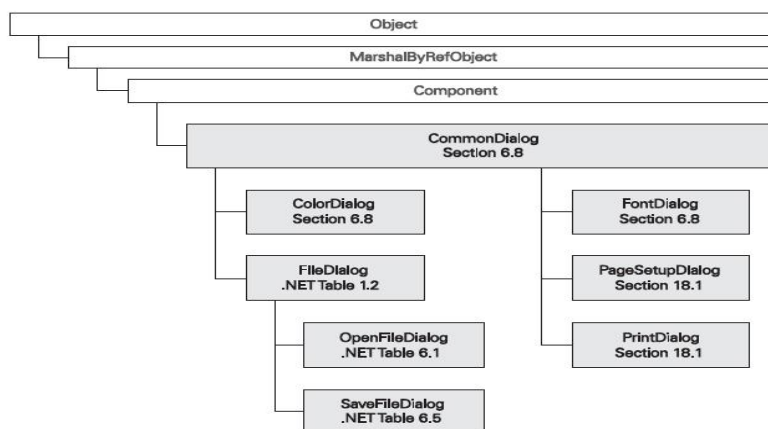
- Là những lớp hộp thoại được thiết kế cho các mục đích sử dụng khác nhau như:
  - Hộp thoại mở file, lưu file
  - Hộp thoại chọn font chữ
  - Hộp thoại chọn màu
  - Hộp thoại chọn thư mục
  - Hộp thoại in ấn
  - ....

Nguyễn Thị Mai Trang 72

72

## Các hộp thoại thông dụng của Windows (tt)

### Common dialogs



Nguyễn Thị Mai Trang

73

73

## Sử dụng các hộp thoại

- Trong thiết kế:
  - Kéo thả vào từ thanh Toolbox
  - Thiết lập các thuộc tính trong cửa sổ Properties
  - Gọi phương thức ShowDialog, kiểm tra kiểu trả về của phương thức ShowDialog để xử lý:
    - DialogResult.OK: chấp nhận thao tác
    - DialogResult.Cancel: hủy thao tác

Nguyễn Thị Mai Trang

74

74

## Sử dụng các hộp thoại (tt)

- Sử dụng code:
  - Khai báo đối tượng của lớp hộp thoại
  - Thiết lập các thuộc tính cho đối tượng
  - Gọi phương thức ShowDialog, kiểm tra kiểu trả về của phương thức ShowDialog để xử lý:
    - DialogResult.OK: chấp nhận thao tác
    - DialogResult.Cancel: hủy thao tác
  - Ví dụ:
 

```
ColorDialog dlg = new ColorDialog();
if(dlg.ShowDialog() == DialogResult.OK) {....}
```

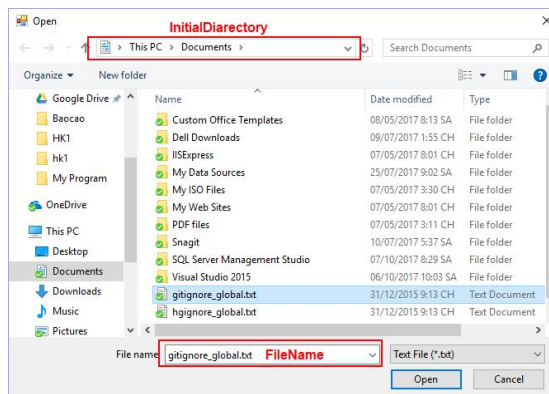
Nguyễn Thị Mai Trang

75

75

## OpenFileDialog

- Hộp thoại cho phép chọn và mở file
- Lớp OpenFileDialog kế thừa từ lớp OpenFileDialog



Nguyễn Thị Mai Trang

76

76

## OpenFileDialog (tt)

- Các thuộc tính cơ bản:
  - Filter: chuỗi quy định loại file được hiển thị trong danh sách các file cho phép người sử dụng chọn.
  - Multiselect: true/false: cho phép chọn nhiều file hoặc chỉ chọn một file, mặc định là false.
  - FileName: file được chọn kiểu string (sử dụng trong trường hợp thuộc tính Multiselect = false)
  - FileNames: danh sách các file được chọn kiểu string (sử dụng trong trường hợp thuộc tính Multiselect = true).

Nguyễn Thị Mai Trang

77

77

## OpenFileDialog (tt)

- Cách sử dụng hộp thoại OpenFileDialog:
  - Kéo biểu tượng OpenFileDialog từ cửa sổ Toolbox vào Form hoặc khai báo và tạo đối tượng bằng code.
  - Thiết lập các thuộc tính cho hộp thoại.
  - Gọi phương thức ShowDialog, truy xuất thuộc tính FileName hoặc FileNames của đối tượng hộp thoại để xử lý.

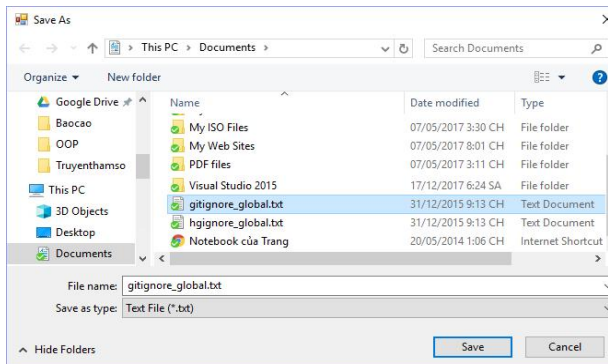
Nguyễn Thị Mai Trang

78

78

## SaveFileDialog

- Tương tự OpenFileDialog, kế thừa từ lớp FileDialog, là hộp thoại cho phép lưu file lên đĩa, không có thuộc tính Multiselect



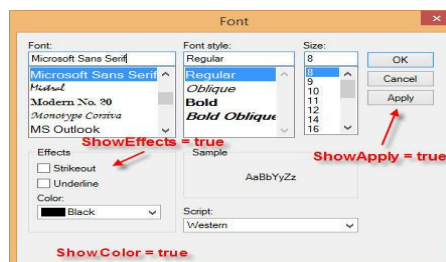
Nguyễn Thị Mai Trang

79

79

## FontDialog

- Hộp thoại cho phép chọn font chữ.



- Các thuộc tính cơ bản:
  - Font: font chữ được chọn.
  - Color: màu chữ được chọn.
  - ShowEffects: true/false - hiển thị/không hiển thị khung Effects.
  - ShowApply: true/false - hiển thị/không hiển thị nút Apply.
  - ShowColor: true/false - hiển thị/không hiển thị combobox chọn màu chữ.

Nguyễn Thị Mai Trang

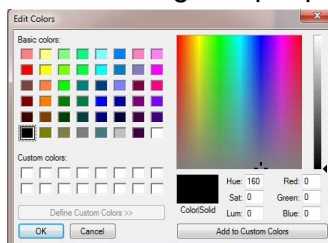
80

80



## ColorDialog

- Hộp thoại cho phép chọn màu
- Các thuộc tính
  - AllowFullOpen: true/false - cho/không cho phép người sử dụng định nghĩa một màu tùy chọn.
  - FullOpen: true/false - cho/không cho phép mở hộp thoại màu dạng đầy đủ.



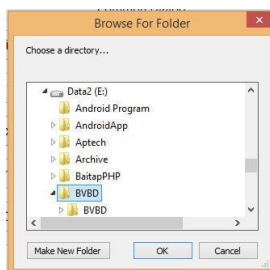
Nguyễn Thị Mai Trang

81

81

## FolderBrowserDialog

- Hộp thoại cho phép chọn thư mục
- Các thuộc tính cơ bản:
  - Description: chuỗi hiển thị trên hộp thoại.
  - SelectedPath: đường dẫn thư mục được chọn.
  - ShowNewFolderButton: True/False: hiển thị/ không hiển thị nút tạo thư mục



Nguyễn Thị Mai Trang

82

82

## 4.11 Ứng dụng dạng SDI - MDI

- **SDI (Single Document Interface)**
  - Tại mỗi thời điểm, chỉ làm việc với một tài liệu
    - NotePad
    - WordPad
    - Paint
    - ...
- **MDI (Multiple Document Interface)**
  - Tại mỗi thời điểm, có thể làm việc với nhiều tài liệu
    - Microsoft Word
    - Microsoft Excel
    - Microsoft PowerPoint
    - ...

## Ứng dụng SDI

- Là loại ứng dụng có một hoặc nhiều form, giữa các form thường có mối liên quan hoặc tương tác với nhau, nhưng không tồn tại loại quan hệ cha-con.
- Các thao tác thường gặp trong ứng dụng có nhiều form là chuyển form và truyền dữ liệu giữa các form.

## Ứng dụng SDI

### • Chuyển qua lại giữa các form:

- Show(): mở form, khi form hiển thị, có thể thao tác với các form khác trong cùng ứng dụng.
- ShowDialog():
  - Mở form dạng modal dialog: người sử dụng phải thao tác với form, không thể thao tác với các thành phần khác của ứng dụng trước khi đóng form.
  - ShowDialog() trả về đối tượng DialogResult, thường dùng để xác nhận hành động của người sử dụng.
- Hide(): ẩn form, giữ nguyên dữ liệu và trạng thái của form trước khi form bị ẩn..

Nguyễn Thị Mai Trang

85

85

## Ứng dụng SDI

- Chuyển form sử dụng Property kiểu đối tượng:

```
//class Form4
Form form1;
//Khai báo Property
public Form Form_1
{
    set { form1 = value; }
}
private void btOpenForm1_Click(
    object sender, EventArgs e)
{
    this.Close();
    form1.Show();
}
```

```
//class Form1
private void btOpenForm4_Click(
    object sender, EventArgs e)
{
    Form4 f = new Form4();
    f.Form_1 = this;
    this.Hide();
    f.Show();
}
```

Nguyễn Thị Mai Trang

86

86

## Ứng dụng SDI (tt)

– Chuyển form sử dụng biến đối tượng static:

```
//class Form5
private void btOpenForm1_Click(
    object sender, EventArgs e)
{
    Form1.form1.Show();
    this.Close();
}
```

```
//class Form1
public static Form1 form1;
private void btOpenForm5_Click(
    object sender, EventArgs e)
{
    form1 = this;
    this.Hide();
    Form5 f = new Form5();
    f.Show();
}
```

Nguyễn Thị Mai Trang

87

87

## Ứng dụng SDI (tt)

- Truyền dữ liệu giữa các form:
  - Sử dụng phương thức khởi tạo:

```
//class Form2
//Khai báo phương thức khởi tạo có tham số
public Form2 (String text):this()
{
    lbText.Text = text;
}
```

```
//class Form1
//Hàm xử lý sự kiện click lên button btOpenForm2
private void btOpenForm2_Click(object sender, EventArgs e)
{
    Form2 f = new
    Form2(txtMessage.Text);
    f.Show();
}
```

Nguyễn Thị Mai Trang

88

88

## Ứng dụng SDI (tt)

– Truyền dữ liệu giữa các form sử dụng Properties:

```
//class Form3
private string message;
//Khai báo Property
public string Message
{
    set { message = value; }
}
//Hàm xử lý sự kiện Form3_Load
private void Form3_Load(
    object sender, EventArgs e)
{
    lblText.Text = message;
}
```

```
//class Form1
private void btOpenForm3_Click (
    object sender, EventArgs e)
{
    Form3 f = new Form3();
    f.Message = txtMessage.Text;
    f.Show();
}
```

Nguyễn Thị Mai Trang

89

89

## Ứng dụng MDI

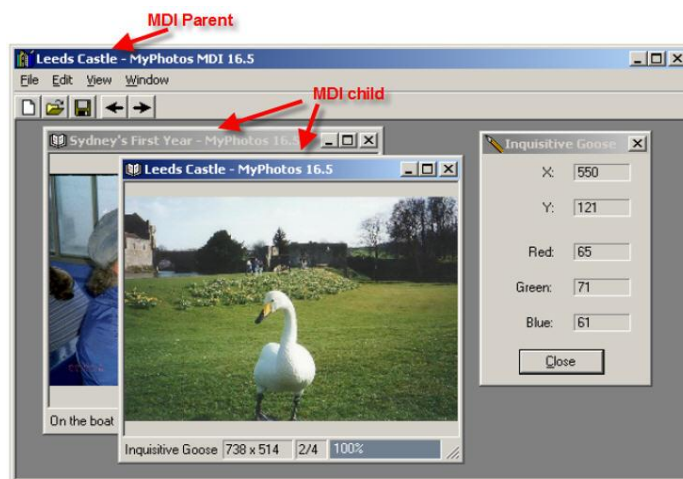
- Là loại ứng dụng mà trong đó có ít nhất một form làm form chính (MDI Form – còn gọi là form cha hay main form).
- Từ form chính có thể gọi mở các form con (child form).
- Các form con khi thực thi nằm trong form cha với nhiều cách sắp xếp đa dạng.
- Ứng dụng MDI gồm có các thành phần cơ bản như:
  - Một form chính của ứng dụng.
  - Ít nhất một form con, các form con này khi mở sẽ nằm trong form cha.
- Menu chính của ứng dụng nằm trên form cha, menu này chứa các chức năng cho phép mở các form con.

Nguyễn Thị Mai Trang

90

90

## Ứng dụng MDI (tt)



Nguyễn Thị Mai Trang

91

91

## Ứng dụng MDI (tt)

- MDI Form:
  - Là form chính chứa các form khác.
  - isMdiContainer = true
- Child Form:
  - form nằm trong MDI Form.
  - MdiParent = form cha.
  - Ví dụ:
    - Form2 frmChild = new Form2();
    - frmChild.MdiParent = this;
    - frmChild.Show ();
  - Trong đó tham chiếu this là form gọi đến Form2.

Nguyễn Thị Mai Trang

92

92

## Ứng dụng MDI (tt)

- Một số thuộc tính của form thường dùng trong ứng dụng MDI:
  - IsMdiChild: true/false - cho biết một form có/không phải là form con.
  - MdiParent: đối tượng Form là form cha của một form con.
  - ActiveMdiChild: đối tượng Form là form con đang hoạt động (active).
  - IsMdiContainer: true/false - xác định form có/không phải là MDI form.
  - MdiChildren: danh sách các form con.

## Ứng dụng MDI (tt)

- Sắp xếp các form con trong ứng dụng MDI:
  - Sử dụng phương thức LayoutMdi
- Duyệt qua các form con:
  - Duyệt danh sách MdiChildren

## Ứng dụng MDI (tt)

- Các bước tạo ứng dụng MDI:
  - Tạo form cha.
  - Thiết kế form con tùy theo yêu cầu của ứng dụng, có thể bổ sung các menu trong các form con này.
  - Bổ sung vào form cha một menu New để có thể gọi các form con.
  - Trộn và sắp xếp các menu của form cha và form con nếu cả form cha và form con đều có sử dụng menu.
  - Gọi form con trong sự kiện click của menu New.
  - Ví dụ: xem tài liệu học tập Lập trình giao diện

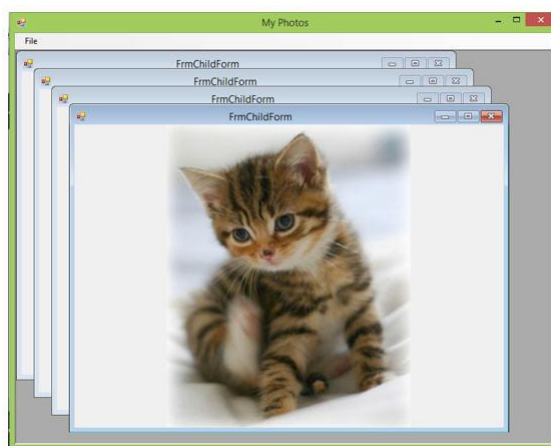
Nguyễn Thị Mai Trang

95

95

## Ứng dụng MDI (tt)

- Ví dụ: xem tài liệu học tập Lập trình giao diện



Nguyễn Thị Mai Trang

96

96