



TRƯỜNG ĐẠI HỌC THỦY LỢI
Khoa CNTT – Bộ môn CNPM

LẬP TRÌNH NÂNG CAO



Giảng viên: Nguyễn Thị Phương Dung

Email: dungntp@tlu.edu.vn

SĐT: 0946 079 903



Lập trình Winform trong C#



Làm quen với winform

- Tạo mới một solution/project
- Cấu trúc thư mục solution
- Giới thiệu giao diện làm việc visual studio
 - Khung thiết kế, viết code
 - Các cửa sổ phụ: solution, toolbox, properties
- Thêm các điều khiển vào form





Start Page - Microsoft Visual Studio

File Edit View Project Debug Team Tools Test Analyze Window Help

Debug x86 Attach...

Start Page

Get Started

Khu vực chứa những bài được mở gần đây

Cần tìm bài hoặc thư mục để mở

Open

Get code from a remote version control system or open something on your local drive.

Checkout from:

- Visual Studio Team Services
- Open Project / Solution
- Open Folder

Recent

The projects, solutions and folders you open locally will appear here.

The remote host for Git repositories and other source control providers will appear on the recent list of other devices you've signed in to.

Tạo bài mới

New project

Search project templates

Recent project templates:

- Console App (.NET Framework) C#
- Windows Forms App (.NET Framework) C#

Create new project...

Tạo mới một chương trình winform

New Project

Recent

Installed

Visual C#

Sort by: Default

WPF App (.NET Framework)	Visual C#
Windows Forms App (.NET Framework)	Visual C#
Console App (.NET Framework)	Visual C#
Class Library (.NET Standard)	Visual C#
Class Library (.NET Framework)	Visual C#
Shared Project	Visual C#
Class Library (Legacy Portable)	Visual C#

Not finding what you are looking for?
[Open Visual Studio Installer](#)

Name: Bai 1

Location: D:\TLU\NNLT nang cao\test\

Solution name: Winform

Framework: .NET Framework 4.6.1

Browse...

☒ Create directory for solution

OK Cancel

Lựa chọn Visual C# để đảm bảo chương trình được viết bằng C#

Lựa chọn Windows Form App để tạo một chương trình chạy bằng giao diện windows

Đặt lại tên cho project của mình (1 project là 1 bài toán nhỏ)

Check vào ô này để hệ thống tự tạo thư mục mới để lưu trữ dự án

Nhớ bấm nút Browse để lựa chọn nơi lưu trữ dự án

Đặt lại tên cho solution (1 solution là 1 chương trình lớn gồm 1 hoặc nhiều bài toán nhỏ)

Cuối cùng bấm nút OK để hoàn tất



Cấu trúc thư mục solution

› TLU › NNLT nang cao › test › WinFormApp

Tên thư mục chứa solution

Name

Tên file solution

WinFormApp.sln

Thư mục con chứa project

Nhapso

› test › WinFormApp › Nhapso

Name

- bin
- obj
- Properties
- Form1.cs
- Form1.Designer.cs
- Nhapso.csproj
- Program.cs

Mỗi thư mục project con sẽ chứa những file mã chương trình có phần mở rộng là **.cs** và các thư mục



Giao diện làm việc

Winform - Microsoft Visual Studio

File Edit View Project Build Debug Team Format Tools Test Analyze Window Help

Quick Launch (Ctrl+Q)

Dung Nguyen Phuong

Toolbox

Properties

Form1

Form1.Designer.cs

Form1.cs [Design]

Form1 System.Windows.Forms.Form

ContextMenuS (none)

ControlBox True

Cursor Default

DoubleBuffer False

Enabled True

Font Microsoft Sans S

ForeColor ControlText

FormBorderStyle Sizable

False

(Icon)

NoControl

False

(Default)

False

0, 0

False

MainMenuStrip (none)

MaximizeBox True

MaximumSize 0, 0

MinimizeBox True

MinimumSize 0, 0

Opacity 100%

Padding 0, 0, 0, 0

RightToLeft No

RightToLeftLay False

ShowIcon True

ShowInTaskbar True

Size 826, 521

SizeGripStyle Auto

StartPosition WindowsDefault

Solution Explorer Team Explorer

Toolbox là cửa sổ chứa các điều khiển

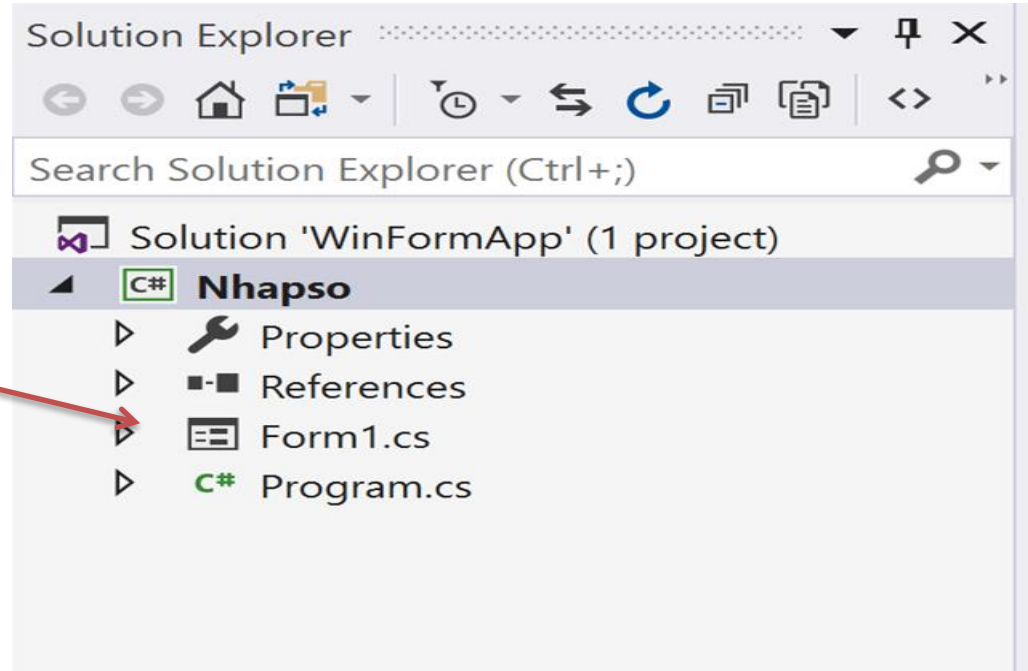
Properties là cửa sổ chứa các thuộc tính và các sự kiện của một điều khiển

Form là giao diện chính của chương trình
Là nơi thiết kế các điều khiển



Cửa sổ solution

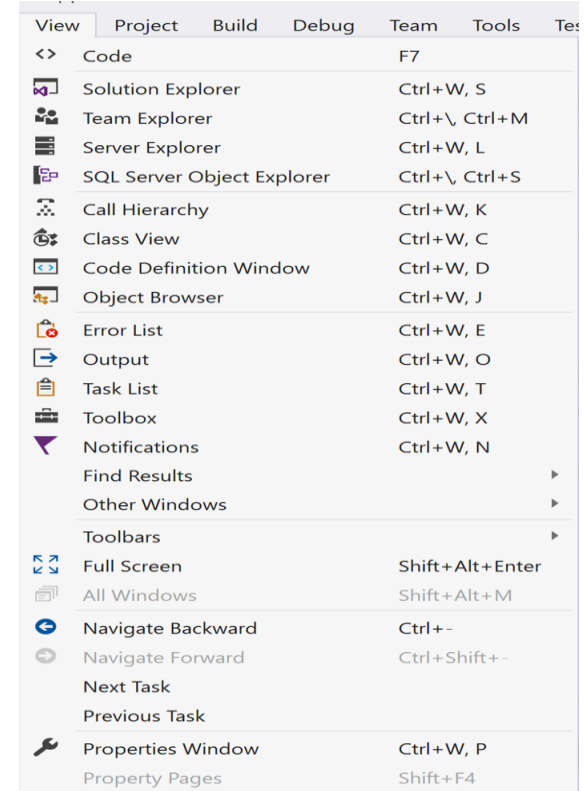
- Là nơi chứa các project trong solution
- Trong mỗi project có các form
- Nháy kép chuột trái vào mỗi form sẽ mở ra khung thiết kế cho form đó





Mở các cửa sổ phụ trợ

- Để mở các cửa sổ bị tắt, lựa chọn thẻ View và chọn cửa sổ cần mở





Thêm các điều khiển vào form

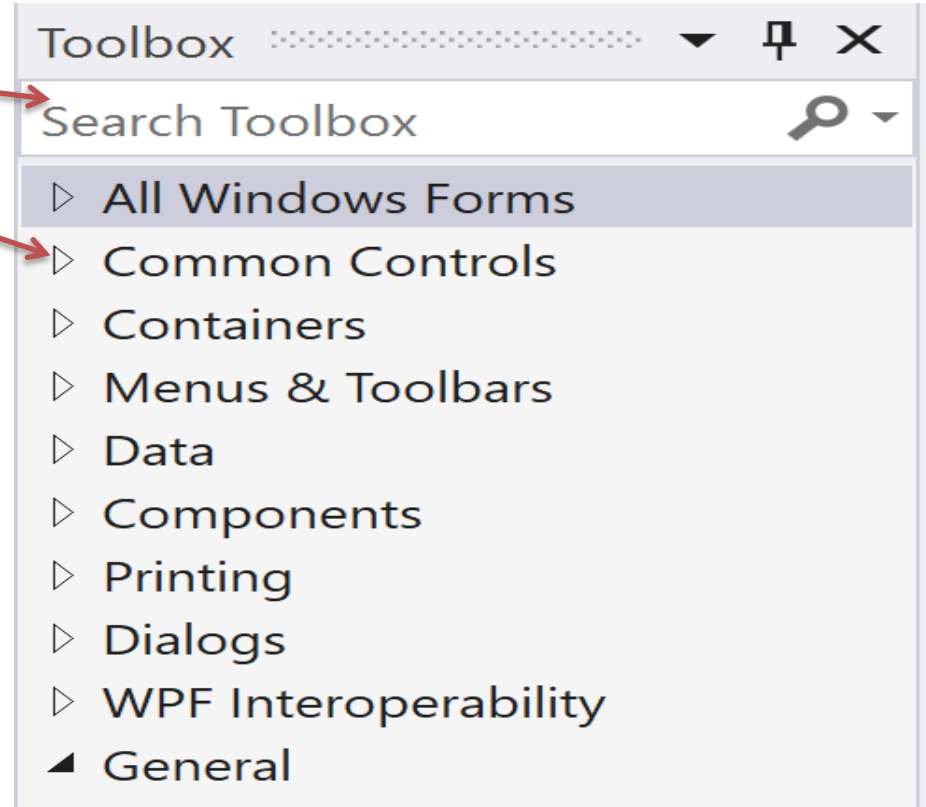
- Mở khung thiết kế
- Lựa chọn đối tượng trên cửa sổ Toolbox
- Kéo thả lên form
- Sắp xếp theo vị trí phù hợp





Cách tìm đối tượng trong Toolbox

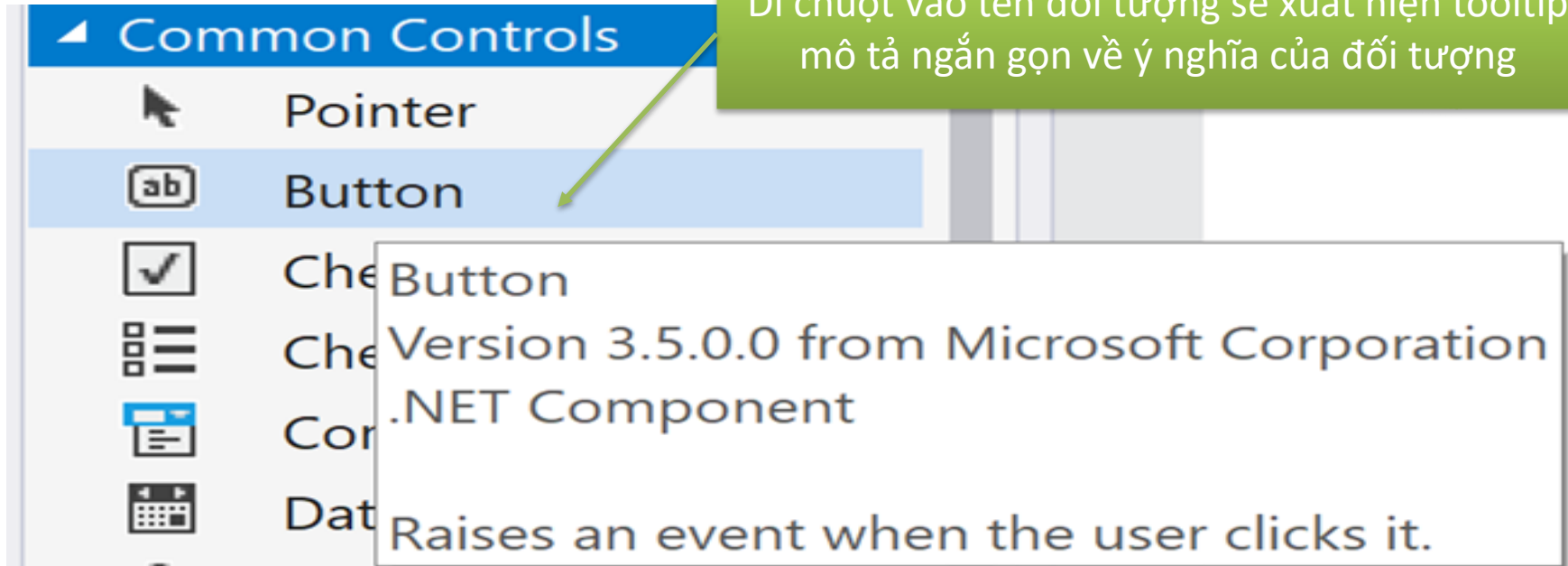
- Gõ vào ô tìm kiếm
- Tìm trong từng nhóm





Tìm hiểu ý nghĩa của từng đối tượng

Di chuột vào tên đối tượng sẽ xuất hiện tooltip mô tả ngắn gọn về ý nghĩa của đối tượng





Một số control thông dụng

Tên control	Mục đích sử dụng
Label	Nhãn hiển thị dữ liệu lên form
TextBox	Ô nhập dữ liệu
Button	Nút bấm
ComboBox	Lựa chọn trong một danh sách xổ xuống
CheckBox	Lựa chọn nhiều giá trị cùng lúc
RadioButton	Chỉ được lựa chọn 1 giá trị trong nhóm





Một số control thông dụng

Tên control	Mục đích sử dụng
DateTimePicker	Chọn ngày tháng năm
NumericUpDown	Lựa chọn số (tránh nhập nhầm)
RichTextBox	Nhập dữ liệu trên nhiều dòng
ListView	Hiển thị danh sách các mục
DataGridView	Hiển thị danh sách dưới dạng bảng





Một số control thông dụng

Tên control	Mục đích sử dụng
MenuStrip	Thực đơn
ContextMenuStrip	Thực đơn cho từng đối tượng, khi bấm chuột phải vào đối tượng đó
ToolTip	Hiển thị ghi chú khi di chuột vào đối tượng





Một số control chứa thông dụng

Tên control	Mục đích sử dụng
GroupBox	Tạo nhóm có tiêu đề
Panel	Tạo nhóm, không có tiêu đề
SplitContainer	Tạo 1 nhóm gồm 2 panel, có thể thay đổi kích cỡ
TabControl	Tạo đối tượng chứa dưới dạng các thẻ
TableLayoutPanel	Thùng chứa ở dạng bảng
FlowLayoutPanel	Thùng chứa tự sắp đối tượng

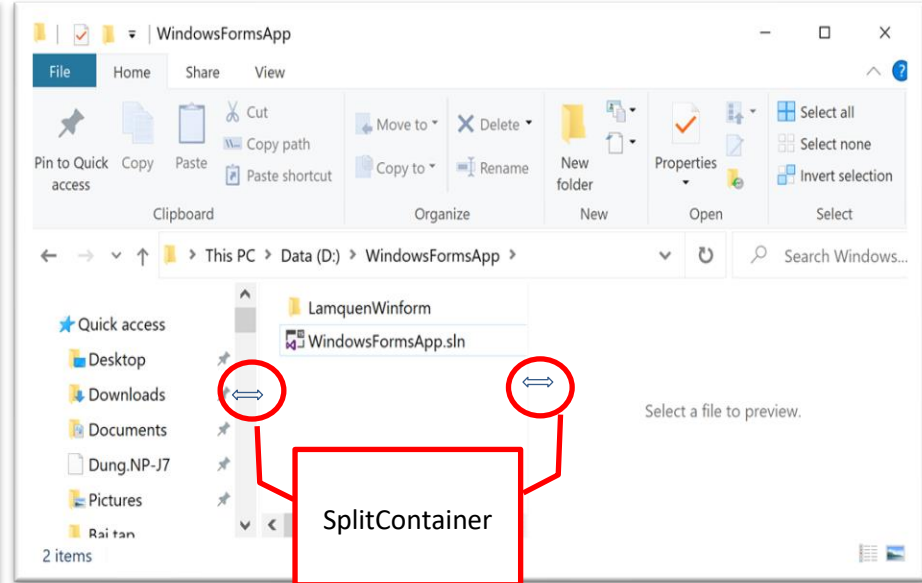
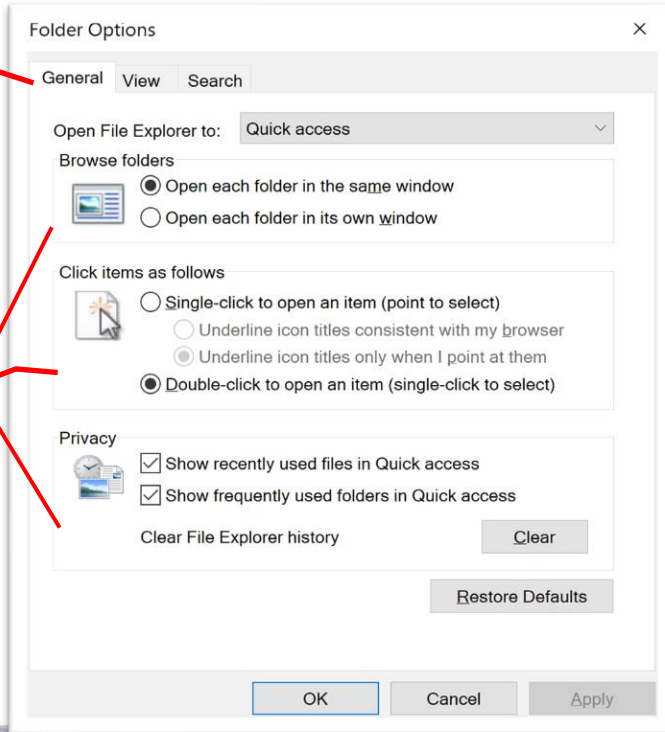




Ví dụ sử dụng thùng chứa

TabControl

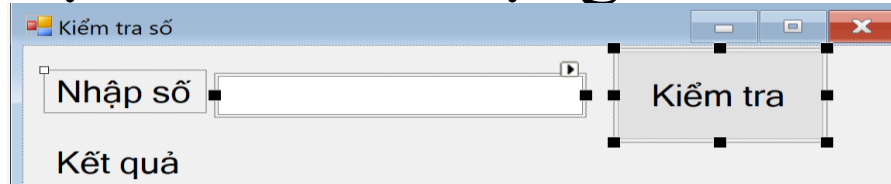
GroupBox





Sắp xếp các đối tượng trên giao diện

- Lựa chọn các đối tượng cần căn lề.



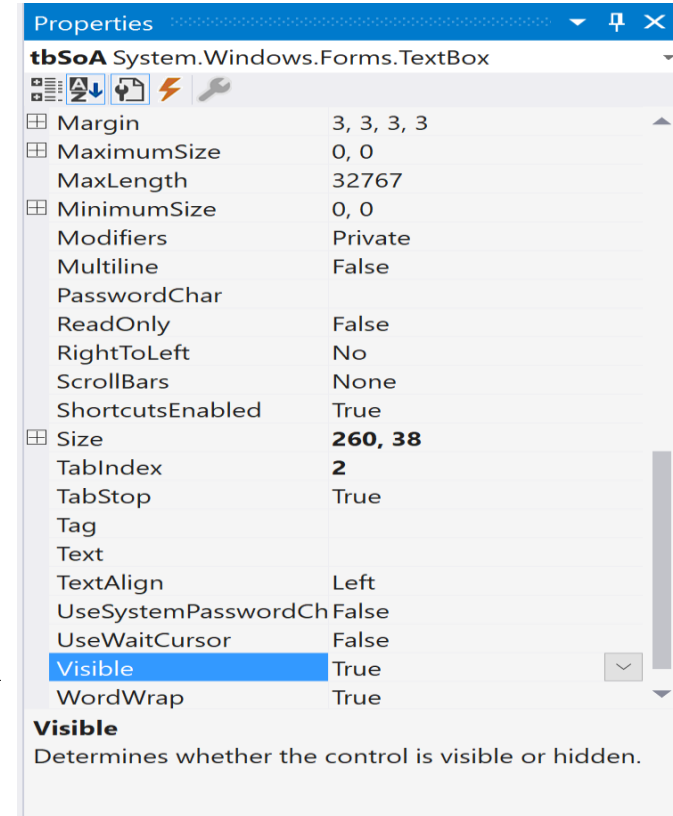
- Sử dụng nút lệnh căn lề trên thanh công cụ





Thiết lập thuộc tính cho các control

- Sử dụng cửa sổ Properties:
 - Chọn Control cần thay đổi thuộc tính.
 - Trên cửa sổ Properties, chọn thuộc tính cần thay đổi và gán lại giá trị thích hợp.
 - Lựa chọn vào thuộc tính nào trong cửa sổ thì phía dưới cửa sổ xuất hiện mô tả ngắn gọn về thuộc tính đó





Thiết lập thuộc tính cho các control

- Sử dụng mã lệnh:
 - Cú pháp: **ControlObject.PropertyName** để truy nhập vào thuộc tính của đối tượng control. Trong đó:
 - **ControlObject** là một thể hiện của Control
 - **PropertyName** là một thuộc tính của control
 - Vd: **tb_Xinchao.Text = "Hello";**





Một số thuộc tính của form

Tên	Ý nghĩa
Name	Tên của Control, đặt theo quy tắc đặt tên biến
Text	Là chuỗi ký tự hiển thị trên thanh tiêu đề của giao diện
ControlBox	Xác định sự xuất hiện của các nút điều khiển của form
MaximizeBox	Xác định nút max có được hiển thị hay không
MinimizeBox	Xác định nút min có được hiển thị hay không
MaximumSize	Xác định kích thước lớn nhất có thể thay đổi
MinimumSize	Xác định kích thước nhỏ nhất có thể thay đổi
StartPosition	Thiết lập vị trí bắt đầu hiển thị của form
BackColor	Thiết lập màu nền của form
BackgroundImage	Thiết lập ảnh nền của form



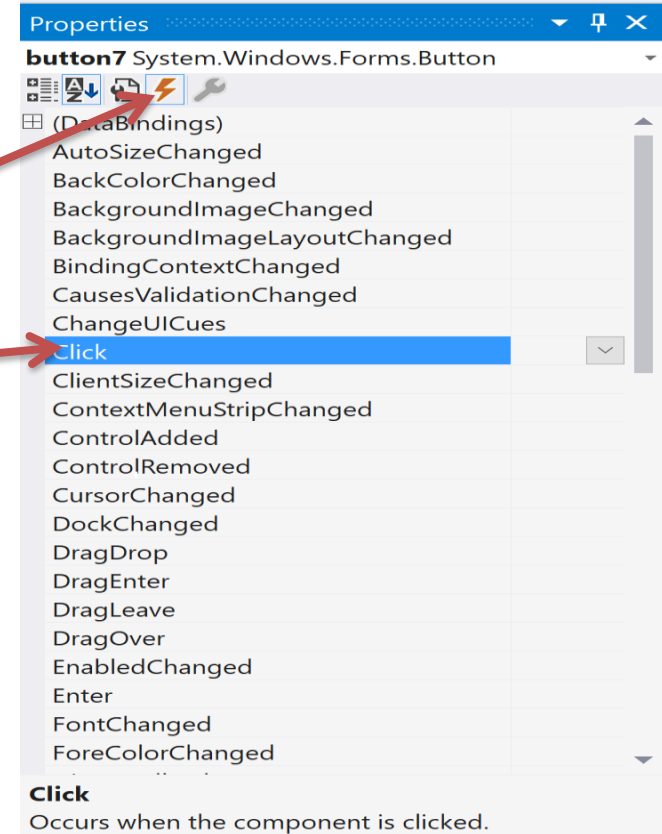
Một số thuộc tính chung của control

Tên	Ý nghĩa
Name	Tên của Control, đặt theo quy tắc đặt tên biến
Text	Là chuỗi ký tự hiển thị lên giao diện
Anchor	Bám Control theo các cạnh của đối tượng chứa, đảm bảo luôn giữ một khoảng cách cho trước với cạnh của đối tượng chứa.
Dock	Bám Control sát theo các cạnh của đối tượng chứa Nếu Dock = Top/ Left/Bottom/Right: bám sát theo 3 cạnh liên quan Nếu Dock = Fill: mở rộng Control ra toàn bộ khoảng trống của đối tượng chứa
Enabled	Để xác định khả năng tương tác của Control
Visible	Xác định khả năng nhìn thấy control vào lúc chạy chương trình.
TabIndex và TabStop	Xác định thứ tự nhận focus của control khi người sử dụng nhấn phím Tab. Nếu không muốn control nhận focus khi người sử dụng dùng phím Tab, gán thuộc tính TabStop của control đó bằng false.



Bắt sự kiện cho các đối tượng

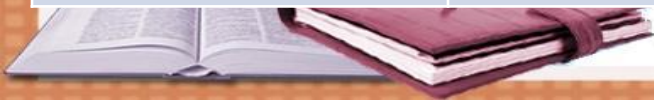
- Lựa chọn đối tượng cần bắt sự kiện trên form
- Mở cửa sổ properties
- Chọn danh sách sự kiện
- Nháy kép chuột vào sự kiện cần bắt
- Cửa sổ viết code được hiện ra với một tên hàm được khai báo mặc định
- Viết code cần thiết vào thân hàm của sự kiện





Một số sự kiện mặc định khi nháy kép chuột trái vào control

Kiểu control	Tên sự kiện mặc định	Ý nghĩa khi chạy chương trình
Form	Load	Sự kiện xảy ra ngay khi form được chạy lên
Button	Click	Sự kiện xảy ra khi bấm chuột lên nút
TextBox	TextChanged	Sự kiện xảy ra khi thay đổi nội dung của textbox
CheckBox	CheckedChanged	Khi thay đổi lựa chọn trong checkbox
ComboBox	SelectedIndexChanged	Khi thay đổi lựa chọn trong combobox
DatagridView	CellContentClick	Khi bấm chuột vào datagridview
NumericUpDown	ValueChanged	Khi thay đổi giá trị trong NumericUpDown





Xóa sự kiện

- Tuyệt đối không xóa trực tiếp đoạn mã lệnh tự sinh cho mỗi sự kiện
- Cần vào cửa sổ Properties, chọn danh sách các sự kiện, chọn vào sự kiện cần xóa và xóa liên kết tới sự kiện tại đây.





Bài tập



Xây dựng một chương trình Window Form để thực hiện:

- Nhập 2 số a và b
- Xuất ra tổng, hiệu, tích, thương của 2 số đó.

Yêu cầu:

- Form không có nút thu nhỏ và phóng to cửa sổ
- Khi chạy lên, form đứng giữa màn hình
- 2 nút Tính và Thoát tự động đứng sát lề bên phải của form khi form thay đổi kích thước
- Các TextBox tự động co giãn theo kích cỡ của form
- Chương trình cho phép sử dụng bàn phím thay vì chuột.
 - Khi ấn nút tab trên bàn phím thì chuyển điều khiển lần lượt từ các control số a, số b, nút Tính và nút Thoát
 - Khi ấn phím enter thì nút Tính được thực hiện
 - Khi ấn phím ESC thì nút Thoát được thực hiện





Một số thuộc tính/ sự kiện đặc trưng
của một số control



Form

Thuộc tính	Ý nghĩa
Name	Tên của form
Text	Tiêu đề của form (hiển thị trên thanh tiêu đề)
BackColor	Thiết lập màu nền của form
BackgroundImage	Chọn ảnh nền cho form
BackgroundImageLayout	Bố cục để ảnh nền của form
FormBorderStyle	Thiết lập kiểu viền của form
Opacity	Thiết lập độ mờ của form (để tỉ lệ càng nhỏ form càng trong suốt)





Form

Thuộc tính	Ý nghĩa
Font	Thiết lập font chữ cho form và những điều khiển kế thừa từ lớp form
ForeColor	Thiết lập màu chữ
ControlBox	Thiết lập hộp điều khiển có xuất hiện hay không (Hộp điều khiển là các nút thu nhỏ, phóng to, đóng cửa sổ ở góc trên bên phải cửa sổ)
MaximizeBox	Cho phép nút phóng to cửa sổ có xuất hiện hay không
MinimizeBox	Cho phép nút thu nhỏ cửa sổ có xuất hiện hay không
StartPosition	Xác định vị trí của cửa sổ khi xuất hiện lần đầu trên màn hình
WindowState	Xác định trạng thái hiển thị form lần đầu





Form

Thuộc tính	Ý nghĩa
AutoScroll	Tự động xuất hiện thanh cuộn khi form bị thu nhỏ, che mất một số control
AutoSize	Chỉ cho phép co giãn cửa sổ đến mức tối thiểu bằng khung chứa các điều khiển hiện tại (không cho phép thu nhỏ đến mức xuất hiện thanh cuộn)
AutoSizeMode	Thiết lập kiểu form có được thay đổi kích cỡ hay không GrowOnly : Được thay đổi GrowAndShrink : Không được thay đổi





Form

Thuộc tính	Ý nghĩa
AutoScaleMode	Tự động thay đổi kích cỡ của control theo tỉ lệ của font chữ hoặc độ phân giải màn hình Font: tỉ lệ theo kích thước font chữ Dpi: tỉ lệ theo độ phân giải màn hình Inherit: tỉ lệ theo lớp cha (nếu có)
Size	Thiết lập kích cỡ cho form
MaximumSize	Thiết lập kích cỡ tối đa cho form
MinimumSize	Thiết lập kích cỡ tối thiểu cho form





Form

Thuộc tính	Ý nghĩa
AcceptButton	Nút mặc định kích hoạt khi ấn phím Enter
CancelButton	Nút mặc định kích hoạt khi ấn phím ESC
ShowIcon	Có hiển thị biểu tượng Icon ở góc trên bên trái cửa sổ hay không
ShowInTaskbar	Thiết lập biểu tượng chương trình khi chạy có xuất hiện ở thanh Taskbar hay không





Button

Sự kiện	Ý nghĩa
Click	<p>Sự kiện mặc định khi nháy kép chuột vào textbox trên màn hình thiết kế.</p> <p>Các lệnh viết trong sự kiện này sẽ được thực hiện khi bấm chuột vào nút trên giao diện chương trình</p>





Label

Thuộc tính	Ý nghĩa
AutoSize	True: Thiết lập chế độ tự động co giãn kích cỡ theo nội dung False: không co giãn kích cỡ theo nội dung, tự động xuống dòng khi nội dung quá dài
TextAlign	Thiết lập kiểu căn lề cho văn bản trong label





TextBox

Thuộc tính	Ý nghĩa
AutoSize	True: Thiết lập chế độ tự động co giãn kích cỡ theo nội dung False: không co giãn kích cỡ theo nội dung, tự động xuống dòng khi nội dung quá dài
ReadOnly	Thiết lập chế độ chỉ đọc cho đối tượng (không được sửa)
TextAlign	Thiết lập kiểu căn lề cho văn bản trong TextBox
PasswordChar	Thiết lập ký tự thay thế khi gõ mật khẩu
Multiline	Cho phép xuất hiện nhiều dòng nếu nội dung trong ô quá dài
ScrollBars	Cho phép xuất hiện thanh cuộn nếu nội dung dài hơn kích cỡ của ô
WordWrap	Thiết lập chữ trong ô tự động xuống dòng





TextBox

Sự kiện	Ý nghĩa
TextChanged	<p>Sự kiện mặc định khi nháy kép chuột vào textbox trên màn hình thiết kế.</p> <p>Các lệnh viết trong sự kiện này sẽ được thực hiện khi gõ văn bản vào Textbox</p>





CheckBox / CheckedListBox / RadioButton

Thuộc tính	Ý nghĩa
Checked	True: Ô lựa chọn đang được đánh dấu False: Ô lựa chọn không được đánh dấu
CheckState	Các trạng thái lựa chọn <ul style="list-style-type: none">- Checked: Có lựa chọn- UnChecked: Không lựa chọn- Indeterminate: Không xác định
ThreeState	Cho phép lựa chọn 3 trạng thái hay không





CheckBox / CheckedListBox / RadioButton

Sự kiện	Ý nghĩa
CheckedChanged	Sự kiện mặc định khi nhấp kép chuột vào đối tượng trên màn hình thiết kế. Thực hiện khi thay đổi trạng thái lựa chọn





ComboBox

Thuộc tính	Ý nghĩa
Items	Tập hợp các lựa chọn
Sorted	Bật/tắt chế độ sắp xếp
DataSource	Thiết lập nguồn dữ liệu dạng DataTable
DisplayMember	Thiết lập trường dữ liệu hiển thị trong danh sách xổ xuống
ValueMember	Thiết lập trường dữ liệu giá trị ẩn tương ứng với dòng dữ liệu của DisplayMember
FormatString	Cho phép thiết lập định dạng hiển thị với một số loại dữ liệu
DropDownHeight	Độ cao khung xổ xuống
DropDownWidth	Độ rộng khung xổ xuống





ComboBox

Sự kiện	Ý nghĩa
SelectedIndexChanged	Sự kiện mặc định khi nhấp kép chuột vào combobox trên màn hình thiết kế. Thực hiện khi thay đổi vị trí lựa chọn





DateTimePicker

Thuộc tính	Ý nghĩa
Format	Kiểu hiển thị
MaxDate	Ngày tháng tối đa
MinDate	Ngày tháng tối thiểu
Value	Giá trị hiện tại





DateTimePicker

Sự kiện	Ý nghĩa
ValueChanged	Sự kiện mặc định khi nháy kép chuột vào DateTimePicker trên màn hình thiết kế. Thực hiện khi thay đổi lựa chọn ngày tháng





DataGridView

Thuộc tính	Ý nghĩa
Columns	Thiết lập các cột hiển thị trong DataGridView
AutoSizeColumnsMode	Thiết lập độ rộng các cột (phủ kín độ rộng của DataGridView nếu để là Fill)
MultiSelect	Cho phép được lựa chọn nhiều dòng hay không
SelectionMode	Thiết lập kiểu lựa chọn
ReadOnly	Cho phép chỉ đọc hoặc được sửa
DataSource	Thiết lập nguồn dữ liệu hiển thị vào DataGridView





DataGridView

Sự kiện	Ý nghĩa
CellContentClick	<p>Sự kiện mặc định khi nháy kép chuột vào DataGridView trên màn hình thiết kế.</p> <p>Thực hiện khi bấm chuột vào nội dung mỗi ô trong DataGridView</p>
CellClick	<p>Sự kiện thực hiện khi lựa chọn từng dòng dữ liệu trong DataGridView (kể cả khi bấm chuột vào vùng trắng, không phải là dữ liệu)</p> <p>Sự kiện này thường được dùng để xác định vị trí dòng đang lựa chọn trên DataGridView (e.RowIndex)</p>





Bài tập



Tạo một form cho phép nhập danh sách sinh viên :

- Nhập thông tin của sinh viên: Mã SV, Họ tên, Giới tính(chỉ cho chọn 1 trong 2 giới), Quê quán (sử dụng Combobox để lựa chọn)
- Có nút Thêm để thêm dữ liệu vào một DataGridView
- Khi bấm vào DataGridView thì hiển thị dữ liệu tại dòng đang bấm lên các Control tương ứng
- Có nút Sửa để cập nhật dữ liệu vừa sửa vào DataGridView
- Có nút Xóa để xóa một dòng dữ liệu đang chọn tại DataGridView

Form titled "Nhập thông tin sinh viên" (Enter student information).

Fields and controls:

- Mã SV: Text box
- Quê quán: Combobox
- Họ tên: Text box
- Lớp: Combobox
- Ngày sinh: Date picker (set to 9/24/2019)
- Khoa: Combobox
- Giới tính: Radio buttons for Nam (Male) and Nữ (Female)
- Thêm: Button

DataGridView below the form:

	Họ tên	Ngày sinh	Giới tính	Quê quán	Lớp	Khoa
*						





Các lớp hộp thoại và cách dùng



Các lớp hộp thoại trong C#

- ColorDialog: hiển thị các màu và trả về màu được chọn
- FontDialog: hiển thị các font và trả về font, kích thước, kiểu chữ được chọn
- OpenFileDialog: cho phép người sử dụng duyệt file và thư mục, sau đó chọn một hoặc nhiều file.
- PageSetupDialog: cho phép người sử dụng lựa chọn các thiết lập liên quan tới cách bố trí trang.





Các lớp hộp thoại trong C#

- `PrintDialog`: cho phép người sử dụng lựa chọn các vấn đề liên quan đến in và gửi tài liệu tới máy in đã chọn.
- `PrintPreviewDialog`: cho phép người sử dụng xem file trước khi in.
- `SaveFileDialog`: cho phép người sử dụng duyệt file và thư mục, sau đó chọn file cần lưu.
- `MessageBox`: cho phép người dùng hiển thị một thông báo với 12 cách hiển thị khác nhau





Phân loại lớp hộp thoại

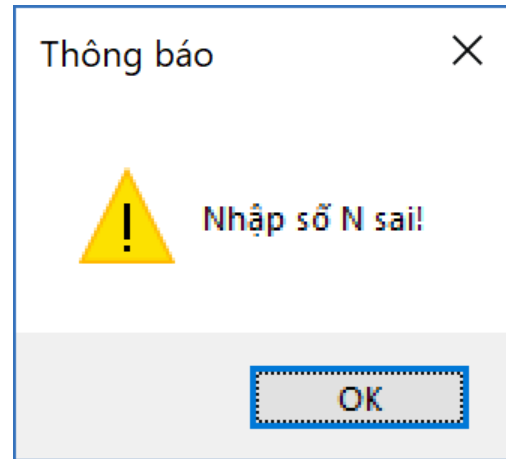
- Hộp thoại được chia thành 2 loại:
 - Modeless - Show()
 - Modal - ShowDialog()





Phân loại lớp hộp thoại

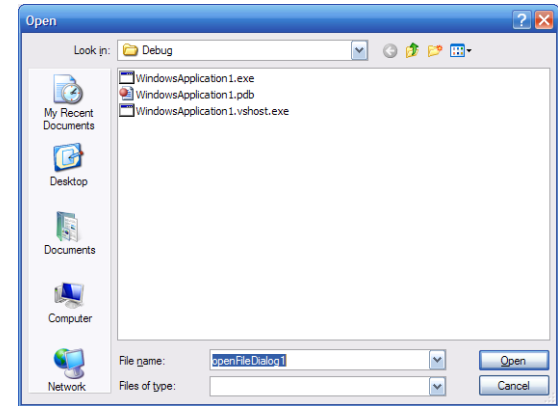
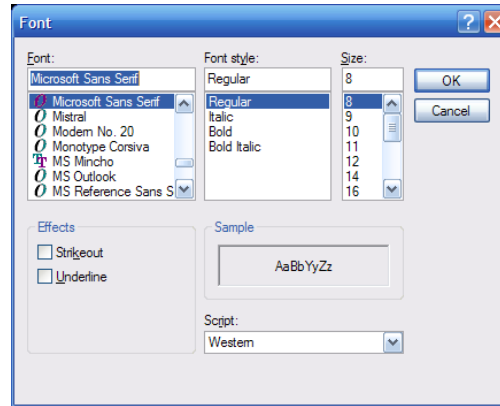
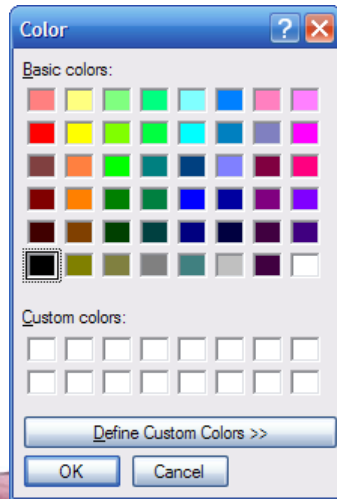
- Modeless - Show(): là loại hộp thoại thông báo, không cần kết quả trả về
- VD:





Phân loại lớp hộp thoại

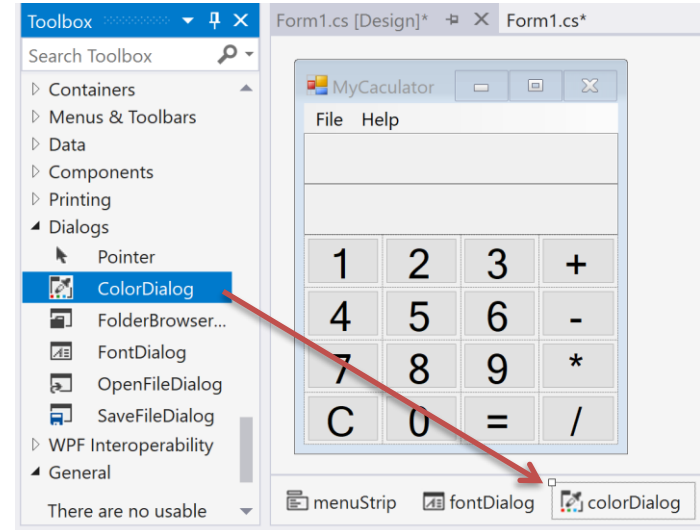
- Modal - ShowDialog(): là hộp thoại cần lấy kết quả trả về như: OK, Cancel,...
- VD:





Sử dụng hộp thoại

- VD: Sử dụng hộp thoại Color
- Trên giao diện, kéo thả công cụ ColorDialog vào form và đặt tên cho đối tượng là ColorDialog





Sử dụng hộp thoại

- Thực hiện việc mở hộp thoại Color ra để lấy màu bằng câu lệnh như sau:

```
if(colorDialog.ShowDialog()==DialogResult.OK)
```

```
    bt1.BackColor = colorDialog.Color;
```





Bài tập



Hãy viết một chương trình mô phỏng chiếc máy tính cầm tay với giao diện như sau:

Yêu cầu:

- Tiêu đề của giao diện ghi: ***“Máy tính của: <Họ và tên SV>”***
- Có một ô TextBox hiển thị kết quả, một ô TextBox hiển thị biểu thức
- Các TextBox có khả năng thay đổi độ dài theo form
- Các nút có kích cỡ bằng nhau và có khả năng thay đổi kích cỡ theo form
- Các nút trên giao diện hoạt động tương tự như máy tính cầm tay.
- Khi bấm vào menu ***“Thay đổi màu sắc”*** thì toàn bộ các nút Button sẽ đổi màu theo màu người dùng chọn
- Khi bấm vào menu ***“Thay đổi font chữ”*** thì toàn bộ font chữ của tất cả các đối tượng sẽ thay đổi theo font mà người dùng chọn

