



Bài 6

WEB FORM

Thời lượng: 240 phút

GV: ThS. PHAN LONG

Email: phanlong92@gmail.com



Mục tiêu

○ Về kiến thức

- Chương này cung cấp cho sinh viên kiến thức lập trình ứng dụng trên Web Form
- Các bước xây dựng ứng dụng Web form, các lớp và đối tượng điều khiển, xử lý sự kiện trên các điều khiển, giúp cho sinh viên sử dụng thành thạo các lớp thư viện của C#.
- Cách thức xây dựng điều khiển server control và custom control



Mục tiêu

○ Về nội dung

Sinh viên có thể vận dụng kiến thức nền tảng về lập trình Web form vào các ứng dụng trong nhiều lĩnh vực



NỘI DUNG

- ❖ Giới thiệu tính năng ASP. Net Web Form
- ❖ Các điều khiển Server Control
- ❖ Lập trình sự kiện trên Server Control
- ❖ Tạo Project ASP.Net Web Form
- ❖ Cấu trúc một Web Form



ASP. Net Web Form

- Là mô hình phát triển ứng dụng web của ASP.Net
- Lập trình hướng sự kiện (Even-driven Programing)
- Tương tự lập trình ứng dụng windows.
- Giúp xây dựng ứng dụng web nhanh chóng, dễ dàng



Giới thiệu tính năng ASP. Net Web Form

ASP.Net Control Categories

Security

Data

Navigation

Web parts

Key Features

Master page

Themes/skins

Localization

Adaptive UI

Application Services APIs

membership / role manager

Site Navigation

Caching

Management



Tạo Project Web Form

ASP.NET Website

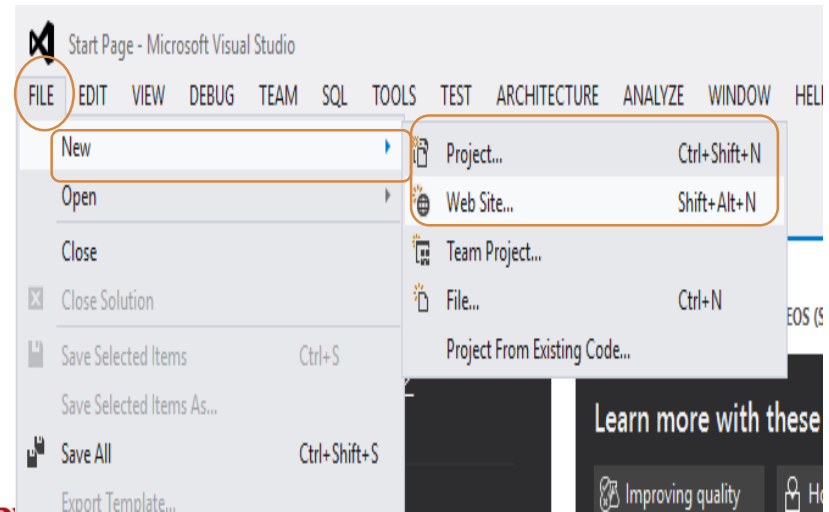
- Thư mục
- Đa ngôn ngữ
- App-Code

ASP.NET Web Application

- Solution
- Đa Project



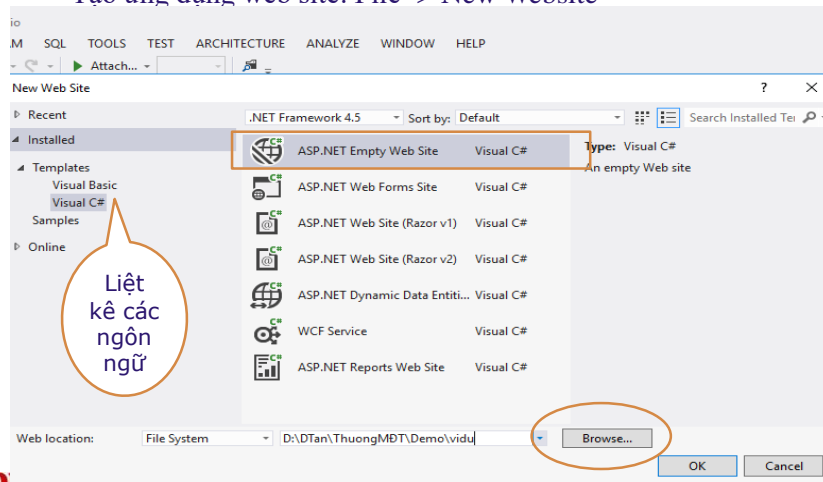
Tạo Project Web Form





Tạo Project Web Form

Tạo ứng dụng web site: File -> New Website



Tạo Project Web Form

Tạo một trang web mới:

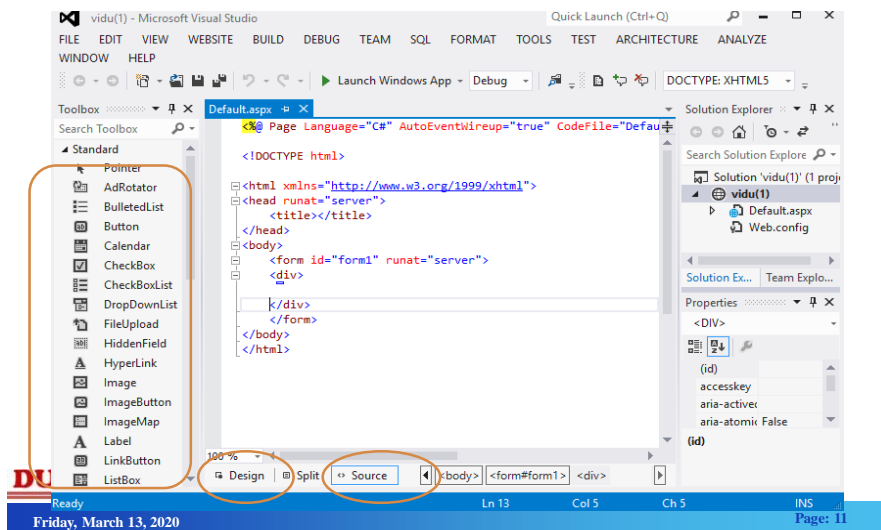
Nhấp phải vào thư mục web -> Add -> Add New Item

Tiếp tục chọn web form -> đặt tên cho trang web (mặc định default.aspx)

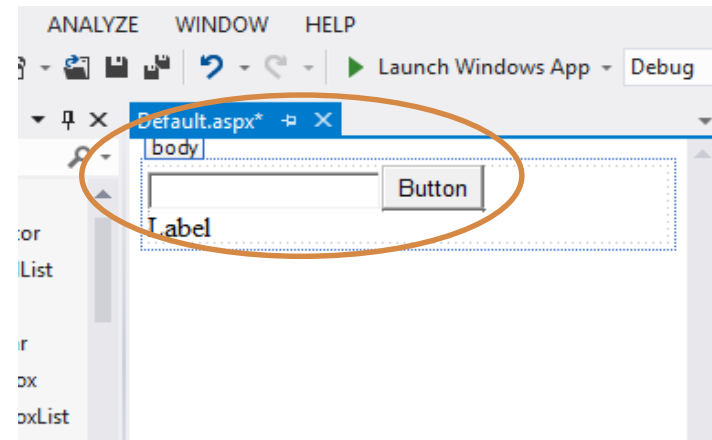


Tạo Project Web Form

Giao diện làm việc



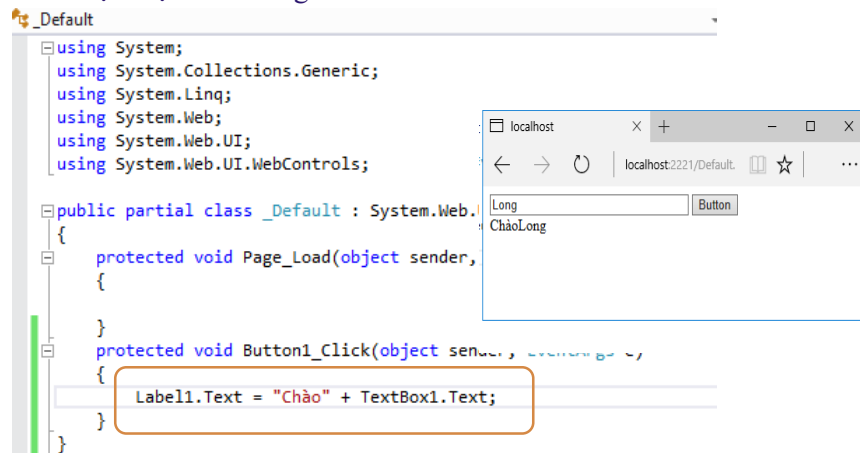
Tạo Project Web Form





Tạo Project Web Form

Tạo một web đơn giản



Lập trình hướng sự kiện

Cách truyền thống

Danh sách các lệnh thực thi tuần tự

Việc kế tiếp xảy ra chính là lệnh tiếp theo trong danh sách

Chương trình được thực thi bởi máy tính

Hướng sự kiện

Các đối tượng có thể kích hoạt sự kiện và các đối tượng khác phản ứng với những sự kiện đó

Việc kế tiếp xảy ra phụ thuộc vào sự kiện kế tiếp

Luồng chương trình được điều khiển bởi sự tương tác User-Computer

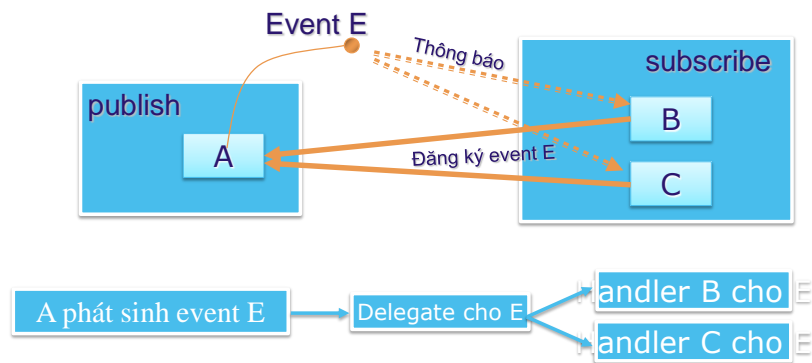
Đợt sự kiện xảy ra

Sử lý và phản hồi



Lập trình hướng sự kiện

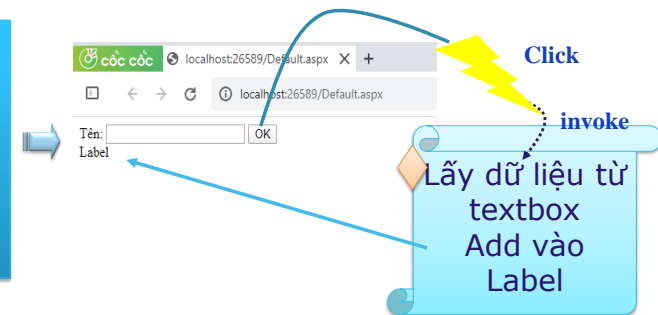
- Trong C#, Event-Driven Programming được thực thi bởi event



Lập trình hướng sự kiện

- Minh họa xử lý trong webform

User nhập text vào textbox → click Button để add chuỗi nhập vào listbox



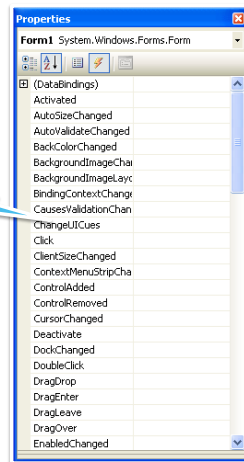
Button đưa ra sự kiện click
Form có event handler cho click của button



Lập trình hướng sự kiện

❖ Các sự kiện

- Mouse move
- Mouse click
- Mouse double-click
- Key press
- Button click
- Menu selection
- Change in focus
- Window activation
- ...



Danh sách
event cho
WebForm



Cấu trúc một Web Form

Một web form bao gồm các thành phần:

- Directives
- Literal Text & HTML tags
- Code Declaration Blocks
- Code Rander Blocks
- Web control
- Server side comments



Cấu trúc một Web Form

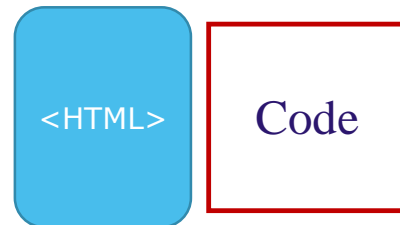
Mã lệnh xử lý có thể đặt trực tiếp trên web form (Code In Line) hoặc tách biệt (code Behind):

Code In-line



Form.aspx

Code Behind



Form.aspx

Form.aspx.cs



Cấu trúc một Web Form

Code Inline

```
<% @ Page Language="C#" %>
<script runat="server">
    void Page_Load(object obj, EventArgs e) {
        lblWelcome.Text = "Welcome to ASP.NET";
    }
</script>
<body>
    <form id="form1" runat="server">
        <div> <asp:Label ID="lblWelcome" runat="server"
            Width="222px"></asp:Label>
        </div>
    </form>
</body>
```



Câu hỏi và Bài tập

1. Trình bày các phương pháp xử lý sự kiện trên các điều khiển Server
2. Phân biệt code behind và Code In Line?
Cho ví dụ.
3. Thảo luận



Cấu trúc một Web Form

Code Behind

- * Mã lệnh được tách biệt;
 - Code Behind là file mã nguồn (C#, VB, ...).
 - Chứa khai báo lớp, có nhiệm vụ xử lý các logic nghiệp vụ của chương trình hay các sự kiện xảy ra khi người dùng tương tác với WebForm
- * Tên của lớp trùng với tên của WebForm
Ví dụ: nếu tên WebForm là **Default** -> Tên Class **Default**



Cấu trúc một Web Form

Code Behind

- * Tất cả các Class xử lý sự kiện trên WebForm đều kế thừa từ lớp **System.Web.UI.Page**
- * Tất cả các Class đều chứa hàm **Page_Load** tự động gọi thực hiện khi WebForm nhận được yêu cầu từ Client
- * Hàm **Page_Load** dùng để khởi tạo nội dung của các control trên WebForm



Cấu trúc một Web Form

Default.aspx

```
<% Page.Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title></title>
</head>
<body>
<form id="form1" runat="server">
<div>

<asp:TextBox ID="TextBox1" runat="server" Width="234px"></asp:TextBox>
<asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Button" /
```

Default.aspx.cs

```
using System.Web.UI.WebControls;

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        Label1.Text = CsharpClass1.Chao(TextBox1.Text);
    }
}
```



Các điều khiển trên Webform

Web forms có thể chứa các kiểu điều khiển (Components) khác nhau:

- Server controls
- HTML controls
- Data controls
- System components



Các điều khiển trên Webform

Web forms có thể chứa các kiểu điều khiển (Components) khác nhau, bao gồm 4 loại:

- Html Server Control
- ASP.Net Server Control
- Validation Control
- Web User Control



WEB SERVER CONTROL

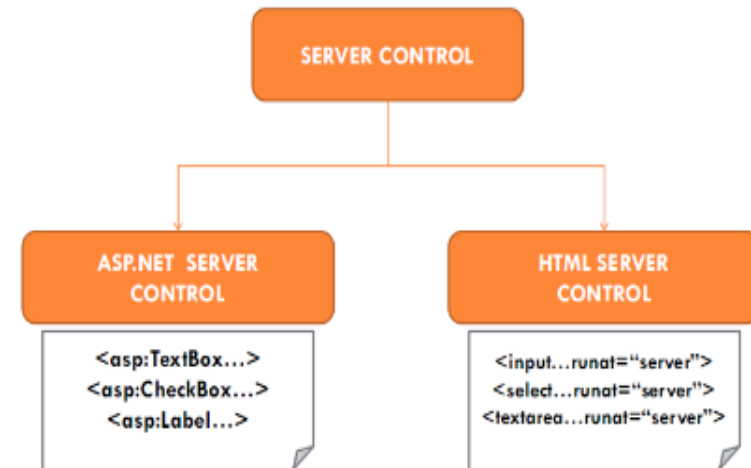
- Thành phần hiển thị nội dung và tương tác với người dùng
- Mỗi web server control bao gồm:
 - Tập các thuộc tính (Properties)
 - Các sự kiện (events)
 - Các phương thức xử lý sự kiện (Methods)
- Web server Control sẽ được tự động chuyển thành mã lệnh HTML tương ứng.

Web Server
Control

HTML



WEB SERVER CONTROL





WEB SERVER CONTROL

- `<asp:Label Text="Label">`
- `<asp:TextBox Text="">`
- `<asp:CheckBox Text="" AutoPostBack="True" Checked="true" />`
- `<asp:RadioButton Text="" AutoPostBack="True" Checked="True" />`
- `<asp:Image ImageUrl="" />`
- `<asp:Button Text="" OnClick="" CommandName="" CommandArgument="">`



WEB SERVER CONTROL

- `<asp:ImageButton ImageUrl="" OnClick="" CommandName="" CommandArgument="" />`
- `<asp:LinkButton Text="" OnClick="" CommandName="" CommandArgument="">`
- `<asp:HyperLink ImageUrl="" NavigateUrl="">`
- `<asp:HiddenField Value="Hello" />`



WEB SERVER CONTROL

- List Controls
 - `<asp:DropDownList>`
 - `<asp:ListBox Rows="5" SelectionMode="Multiple">`
 - `<asp:CheckBoxList RepeatDirection="Horizontal" RepeatLayout="Flow">`
 - `<asp:RadioButtonList RepeatDirection="Horizontal" RepeatLayout="Flow">`
 - `<asp:BulletedList BulletImageUrl="s" DisplayMode="Text">`



WEB SERVER CONTROL

- Các thuộc tính hay dùng
 - Items
 - SelectedItem
 - AutoPostBack="true"
 - DataSource, DataMember, DataTextField, DataValueField
- Sự kiện
 - SelectedIndexChanged



WEB SERVER CONTROL

- `<asp:Panel`
`GroupingText="Gender"></asp:Panel>`
 - Sử dụng để nhóm các controls
- `<asp:MultiView ActiveViewIndex="1">`
`<asp:View></asp:View>`
`<asp:View></asp:View>`
- `</asp:MultiView>`
 - Tạo nhiều khung nhìn, mỗi thời điểm chỉ thấy 1 khung nhìn
- `<Wizard>`
 - Tạo giao diện nhiều bước



CÂU HỎI

1. Thế nào là lập trình hướng sự kiện?
2. Webform là gì?
3. Phân biệt Code behind và Code inline
4. Nêu các thuộc tính cơ bản của các WebServer Control





BÀI TẬP

BÀI TẬP: Sử dụng web server control tạo giao diện sau

The screenshot shows a web browser window with the address bar displaying 'localhost:1584/WebSite2/Default.aspx'. The page contains a form with the following elements:

- A calendar control showing March 2011.
- A text box labeled 'Mã sinh viên:'.
- A text box labeled 'Họ và tên:'.
- A text box labeled 'Ngày sinh:' with the value '09-03-2011'.
- A group box labeled 'Giới tính:' containing two radio buttons: 'Nam' (selected) and 'Nữ'.
- A dropdown menu labeled 'Quốc tịch' with 'Việt Nam' selected.
- A section labeled 'Sở thích:' with three checkboxes: 'Đọc sách' (unchecked), 'Đi du lịch' (checked), and 'Chơi thể thao' (unchecked). There is also a 'Khác' checkbox.
- A section labeled 'Hình ảnh:' with a 'Choose File' button and the text 'No file chosen'.
- An 'OK' button.

DU

Friday, March 13, 2020

Page: 35



TRUYỀN DỮ LIỆU GIỮA CÁC TRANG WEB FORM

Đối tượng Context là đối tượng của lớp HttpContext, cung cấp truy cập đến toàn bộ ngữ cảnh hiện hành. Bạn có thể sử dụng đối tượng này để chia sẻ thông tin giữa các trang Web

Phương pháp 1:

Sử dụng Context.Handler

Ví dụ 1: Khai báo thuộc tính public tên là Value trả về dữ liệu nhập vào trong text box trên Web form Context1

DUY TAN
edu.vn

Friday, March 13, 2020

Page: 36



TRUYỀN DỮ LIỆU GIỮA CÁC TRANG WEB FORM

Phương pháp 2:

Sử dụng thuộc tính tập hợp Context.Items lưu trữ và trả về các giá trị hay đối tượng dữ liệu value với tên khóa key

Lưu giá trị vào tập hợp Context.Items:

Context.Items["key"] = value/object;

Nhận giá trị từ tập hợp Context.Items:

Context.Items["key"]



TRUYỀN DỮ LIỆU GIỮA CÁC TRANG WEB FORM

Phương pháp 3:

- Sử dụng thuộc tính tập hợp Request.Form, là tập hợp các giá trị từ các điều khiển trên Web form khi submit form theo phương thức post

string Request.Form[string *name*]

hay string Request.Form[int *index*]

- để nhận giá trị chuỗi của các điều khiển trên Web form có tên là name, nếu không tồn tại điều khiển với tên name trên Web form, hàm trả về chuỗi rỗng ""



BÀI TẬP

Bài tập

Tạo Web form Menu.aspx, kích vào các liên kết
LinkButton, sẽ chuyển đến Web form Gianhang.aspx, tùy
thuộc người dùng chọn gian hàng nào, sẽ hiển thị:
Tên gian hàng đó trong một điều khiển Literal hay Label
trên Web form hiện hành
Tên gian hàng đó trên một Web form khác

GIAN HÀNG

[Ao quan](#)

[Giay dep](#)

[Tui xach](#)



LINK THAM KHẢO

<https://www.youtube.com/watch?v=aBbfgKYG70>



Kết thúc