

BÀI GIẢNG LẬP TRÌNH WEB

Chương 3. LÀM VIỆC VỚI TRANG ASP.NET

Nguyễn Thành Thủy

Trường Đại học Kinh tế - ĐH Đà Nẵng

e-Mail: thuynt@due.edu.vn

NỘI DUNG

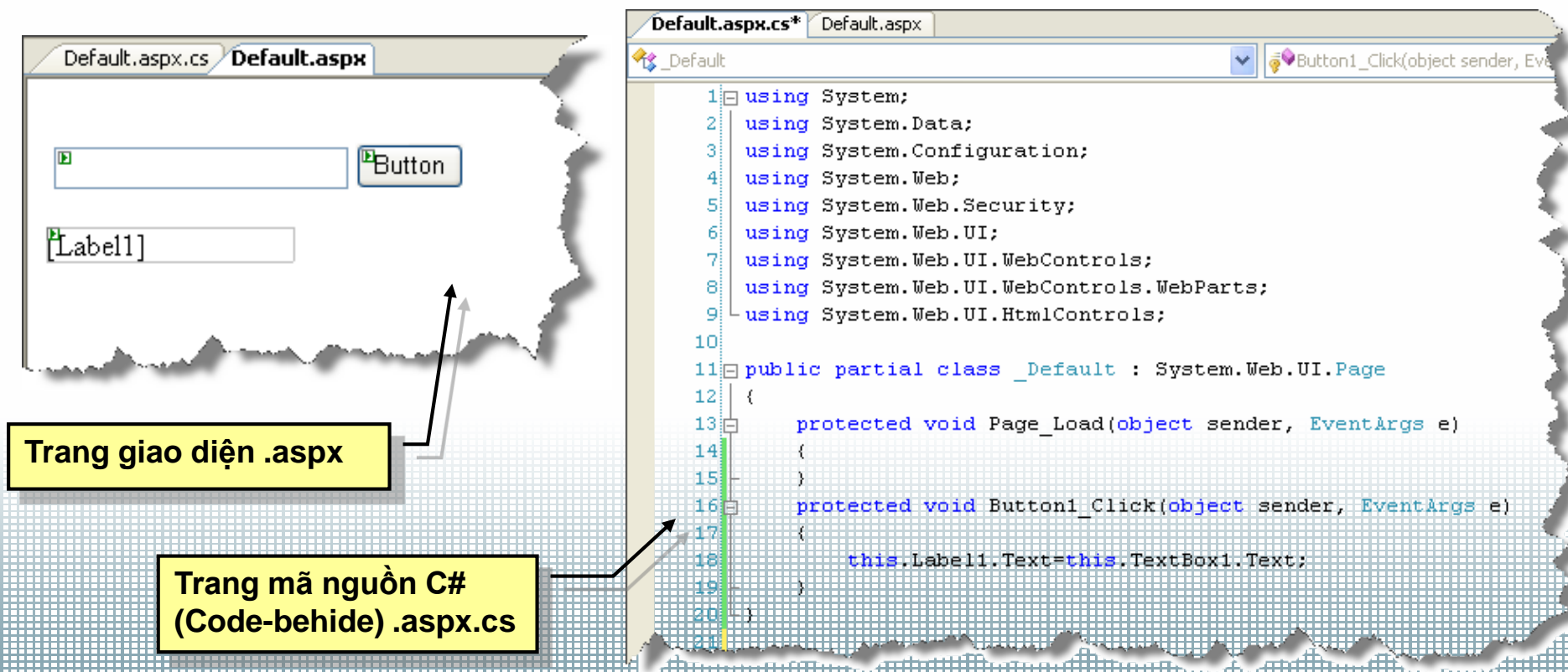
- ❑ Mô hình mã của trang ASP.NET
- ❑ Cấu trúc một trang WebForm
- ❑ Phân loại tập tin trong ASP.Net
- ❑ Chế độ gỡ rối Debugging
- ❑ Các sự kiện trong ASP.NET
- ❑ Tập tin Web.config
- ❑ Tập tin Global.asax



Mô hình mã của trang ASP.NET

❑ Phương pháp sử dụng trang Code-behinde

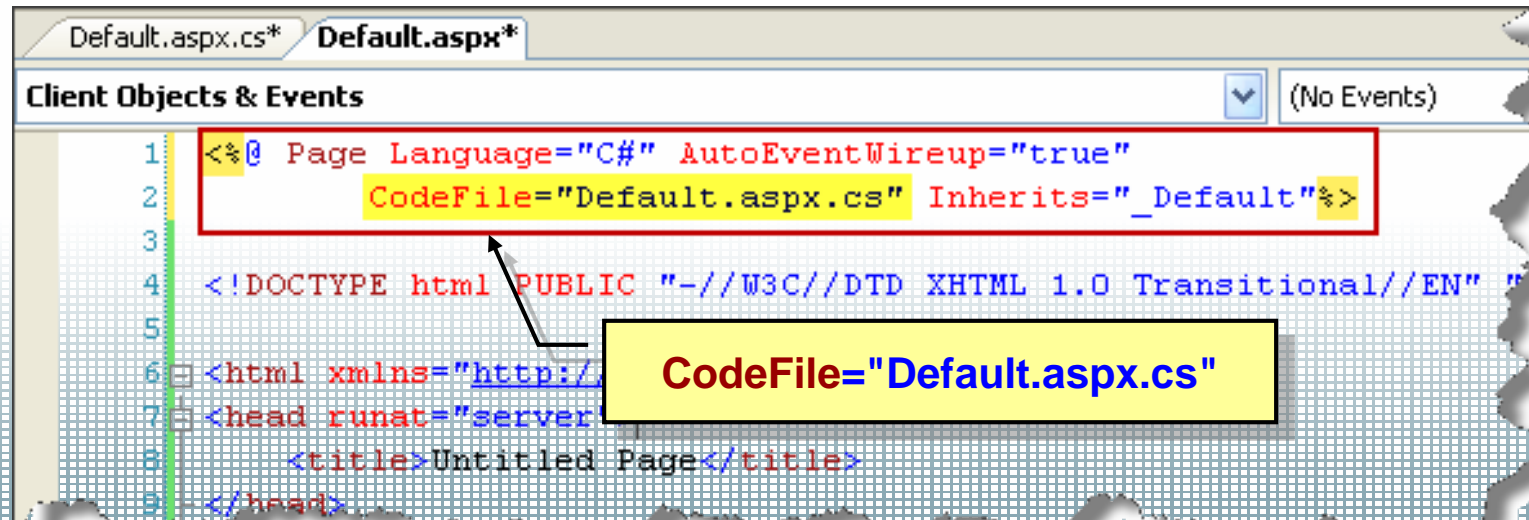
- ❑ Trang ASP.NET được chia thành 2 phần chính, ứng với phần giao diện (**.aspx**) và phần mã nguồn (Code-behinde) C#/VB.NET (**.aspx.cs/.aspx.vb**)



Mô hình mã của trang ASP.NET (2)

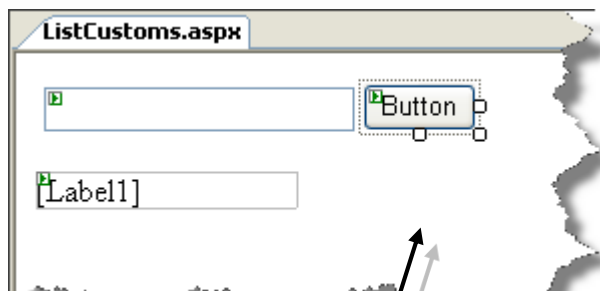
❑ Trang Code-behind:

- ❑ Là file mã nguồn (C#/ VB.net) chứa các khai báo lớp có chức năng xử lý các nghiệp vụ hay các sự kiện xảy ra khi người dùng tương tác với WebForm.
- ❑ Tên của file Code-behind trùng với tên của trang giao diện và có phần mở rộng là **aspx.cs** (**aspx.vb**)
- ❑ Trang code-behind tương ứng được khai báo chỉ dẫn trong trang giao diện **.aspx** với thẻ thuộc tính: **CodeFile**

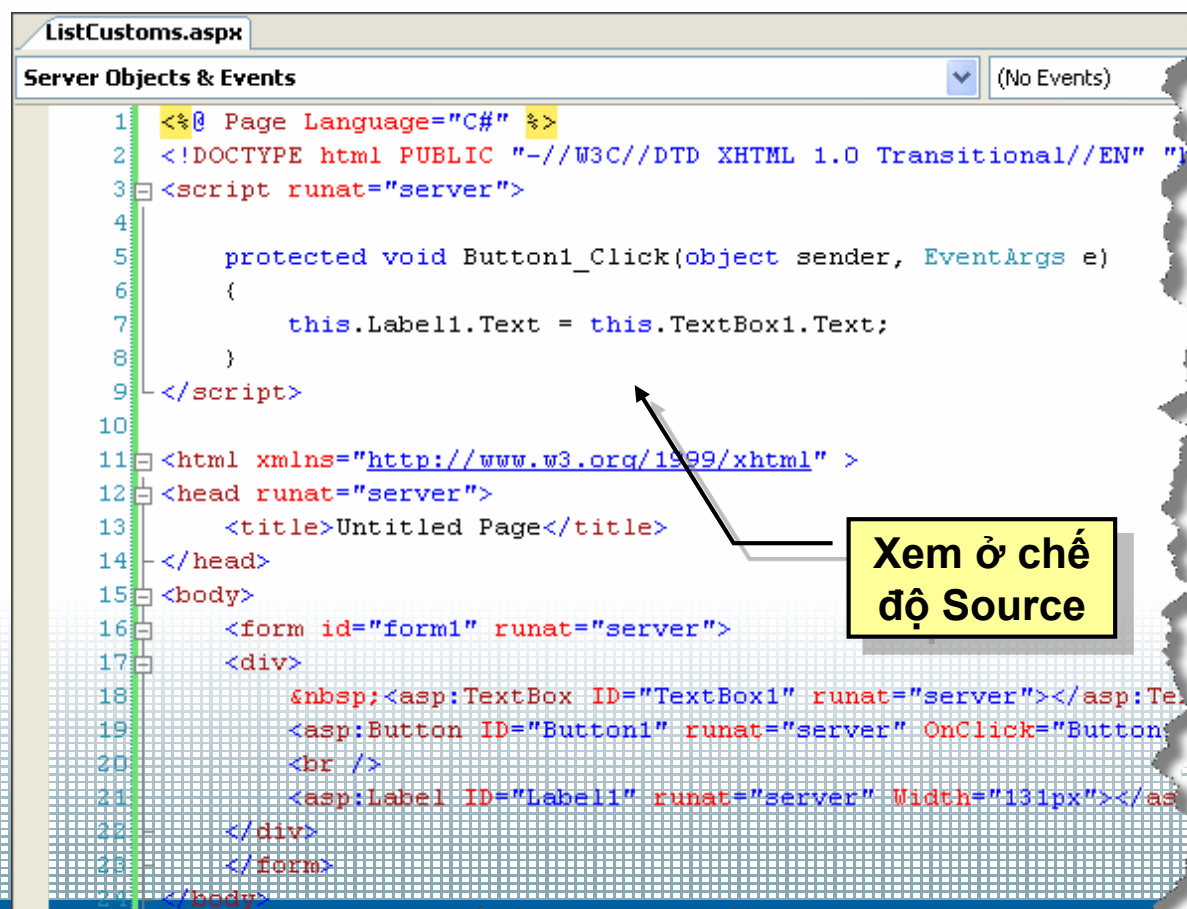


Mô hình mã của trang ASP.NET (3)

- ❑ Phương pháp không sử dụng trang **Code-behind**: người thiết kế có thể đặt cả phần giao diện và phần mã nguồn trong cùng một trang **ASPX**



Xem ở chế độ Design



```
1 <%@ Page Language="C#" %>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "
3 <script runat="server">
4
5     protected void Button1_Click(object sender, EventArgs e)
6     {
7         this.Label1.Text = this.TextBox1.Text;
8     }
9 </script>
10
11 <html xmlns="http://www.w3.org/1999/xhtml" >
12 <head runat="server">
13     <title>Untitled Page</title>
14 </head>
15 <body>
16     <form id="form1" runat="server">
17     <div>
18         &nbsp;<asp:TextBox ID="TextBox1" runat="server"></asp:Te
19         <asp:Button ID="Button1" runat="server" OnClick="Button
20     <br />
21         <asp:Label ID="Label1" runat="server" Width="131px"></as
22     </div>
23 </form>
24 </body>
```

Xem ở chế độ Source

Cấu trúc một trang WebForm

- ❑ Các thành phần của một trang WebForm
 - ❑ Derectives (*Chỉ dẫn*)
 - ❑ Code **Declaration** Blocks (*Khối khai báo các hàm/sự kiện*)
 - ❑ Code **Render** Blocks
 - ❑ Web **Controls** (*Các điều khiển*)
 - ❑ Server-side comments (*Các đặc tả về Server-side*)
 - ❑ Literal Text and HTML tags (*Nội dung tĩnh và các thẻ HTML*)

Cấu trúc một trang WebForm (2)

□ Derectives

- Chứa các chỉ thị cho biết cách thức biên dịch của WebForm.
- Được khai báo trong cặp dấu: **<%@ ... %>** và có thể đặt ở bất kỳ vị trí nào trong trang WebForm.

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

Cấu trúc một trang WebForm (3)

❑ Code Declaration Blocks

- ❑ Là khối khai báo các phương thức hoặc các hàm xử lý sự kiện
- ❑ Được sử dụng trong phương pháp **không sử dụng trang Code-behind**

```
<script runat="server">
    protected void Page_Load()
    {
        this.TextBox1.Text = Request.Form.Get("TextBox1");
    }
</script>
```


Cấu trúc một trang WebForm (4)

❑ Code Render Blocks

- ❑ Là khởi khai báo các lệnh được xử lý khi trang web được load lên.
- ❑ Khối lệnh được đặt trong cặp dấu **<% ... %>**

```
<%  
    string Msg = "Hello World";  
%>  
<%=Msg%>
```

Cấu trúc một trang WebForm (5)

□ Web Control

- Bao gồm các thẻ lệnh khai báo các điều khiển ứng dụng Web
- Được khai báo trong cặp dấu:

<form runat="server"> ... </form>

```
<form id="form1" runat="server">  
    <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>  
    <asp:Button ID="Button1" runat="server" Text="Button" />  
</form>
```

Cấu trúc một trang WebForm (6)

❑ Server-side comments

- ❑ Vùng thể hiện các nội dung ghi chú của người thiết kế.

- Nội dung Comment sẽ được bỏ qua khi trang web được biên dịch.

- ❑ Sử dụng một trong hai dạng sau:

- **HTML Comment:**

<!-- Comment -->

- **ASP.NET Comment:**

<%-- Comment --%>

```
<%-- Khởi khai báo các Web Control --%>
<form id="form1" runat="server">
    <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
    <asp:Button ID="Button1" runat="server" Text="Button" />
</form>
<!-- Khởi khai báo các Code Render -->
<%
    string Msg = "Hello World";
%>
<%=Msg%>
```

Cấu trúc một trang WebForm (7)

□ Literal Text and HTML tags

```
<%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://
xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head runat="server">
  <title>Hello World</title>
</head>
<script runat="server">
  protected void Page_Load()
  {
    this.TextBox1.Text = Request.Form.Get("TextBox1");
  }
</script>
<body>
  <span style="color: #cc0000"><strong>Chào các bạn !!!<br />
    Đây là trang web ASP.NET đầu tiên của tôi.</strong></span>
  <br />
  <%= "Bây giờ là: " + DateTime.Now.ToString() %>
  <form runat="server">
    <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
    <asp:Button ID="Button1" runat="server" Text="Button" />
  </form>
</body>
</html>
```

Cấu trúc một trang WebForm (8)

Trường hợp tích hợp giao diện và Code-behinde trong một trang

```
1 <%@ Page Language="C#" %>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1
  transitional.dtd">
3 <html>
4 <head runat="server">
5     <title>Hello World</title>
6 </head>
7 <script runat="server">
8     protected void Page_Load()
9     {
10         this.TextBox1.Text = Request.Form.Get("TextBox1");
11     }
12 </script>
13 <body>
14     <!-- Comment here-->
15     <form id="form1" runat="server">
16         <asp:TextBox ID="TextBox1" runat="
17         <asp:Button ID="Button1" runat="server" Text="Button" />
18     </form>
19     <%=Request.Form.Get("TextBox1")%>
20 </body>
21 </html>
```

Page Derectives

Literal Text and HTML tags

Code Declaration Blocks

Server-side comments

Web Controls

Code Render Blocks

Cấu trúc một trang WebForm (9)

Trường hợp mã giao diện và Code-behind được đặt ở hai file khác nhau

Page Directives

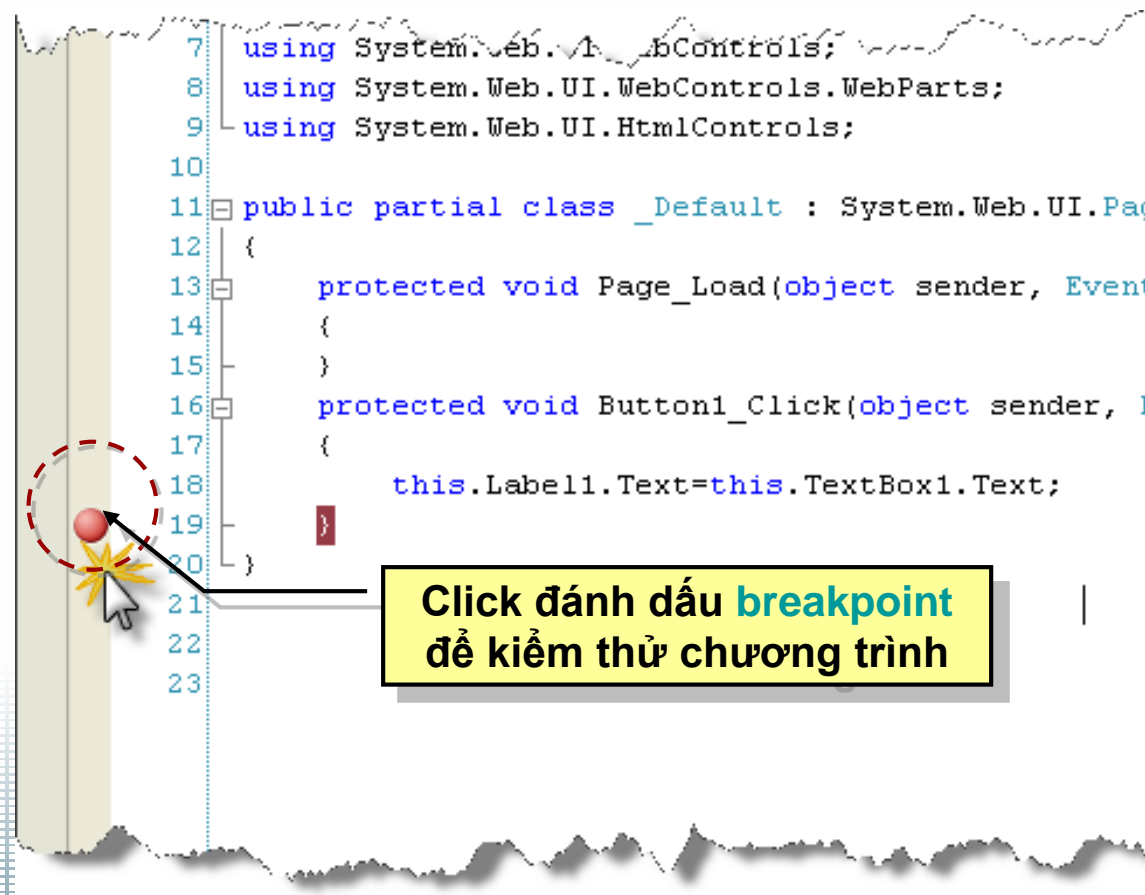
```
1 <%@ Page Language="C#" AutoEventWireup="true" CodeFile=
2 "ListProducts.aspx.cs" Inherits="ListProducts" %>
3
4 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
5 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
6
7 <html xmlns="http://www.w3.org/1999/xhtml" >
8 <head runat="server">
9 <title>Untitled Page</title>
10 </head>
11 <body>
12 <%-- Comment here--%>
13 <form id="form1" runat="server">
14 <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
15 <asp:Button ID="Button1" runat="server" Text="Button" />
16 </form>
17 <%=Request.Form.Get("TextBox1") %>
18 </body>
19 </html>
```

Phân loại tập tin trong ASP.NET

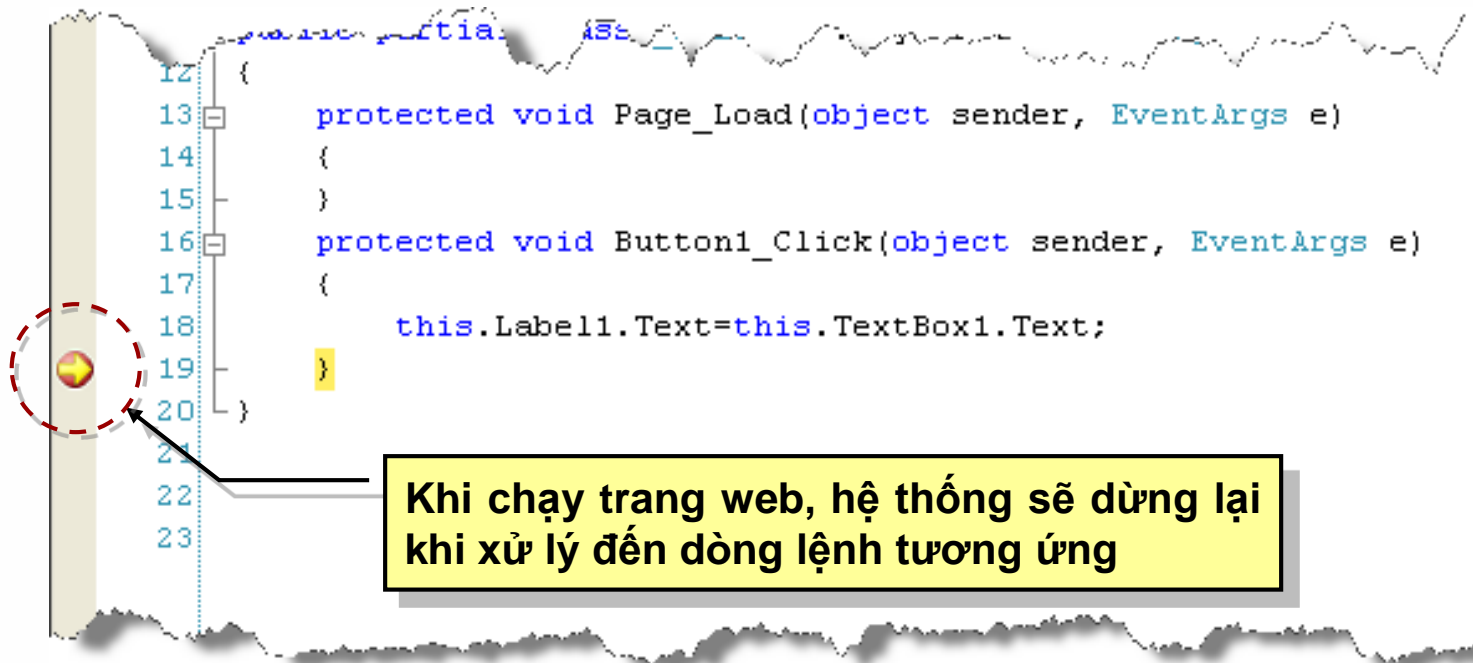
ASP.NET	Diễn giải
.asax	Tập tin global.asax , là tập tin quản lý các sự kiện ứng dụng (application), sự kiện phiên (session) và các sự kiện khác khi có yêu cầu đến 1 trang web trong website
.ascx	Các điều khiển do người dùng tự tạo được lưu trữ với phần mở rộng là ascx.
.asmx	Tập tin Web Service của ứng dụng ASP.Net
.aspx	Phần mở rộng mặc định của trang ASP.Net
.config	Tập tin cấu hình ứng dụng theo định dạng XML. Web.config chứa hầu hết các cấu hình của ứng dụng
.cs / .vb	Tập tin mã nguồn viết theo ngôn ngữ C# / VB.NET
.js	Tập tin mã nguồn của JavaScript

Chế độ gỡ rối Debugging

- ❑ **Debugging** (gỡ rối) cho phép người thiết kế có thể kiểm thử chương trình ở từng dòng lệnh.



Chế độ gỡ rối Debugging (2)



Chế độ gỡ rối Debugging (3)

Giá trị của các biến/đối tượng khi trang web được xử lý đến đoạn code tương ứng

Locals		
Name	Value	Type
this	{ASP.default_aspx}	_Default {ASP.default_aspx}
sender	{Text = "Button"}	object {System.Web.UI.WebControls.Button}
e	{System.EventArgs}	System.EventArgs

Click nút **Stop** để
dừng việc kiểm thử



Các sự kiện trong ASP.NET

- ❑ **Giới thiệu về sự kiện (Event)**
- ❑ **Các sự kiện chính của ứng dụng web**
 - ❑ Các sự kiện chính của **Application** và **Session**
 - ❑ Các sự kiện của **WebForm**
 - ❑ Các sự kiện của **Server Control**

Các sự kiện trong ASP.NET (2)

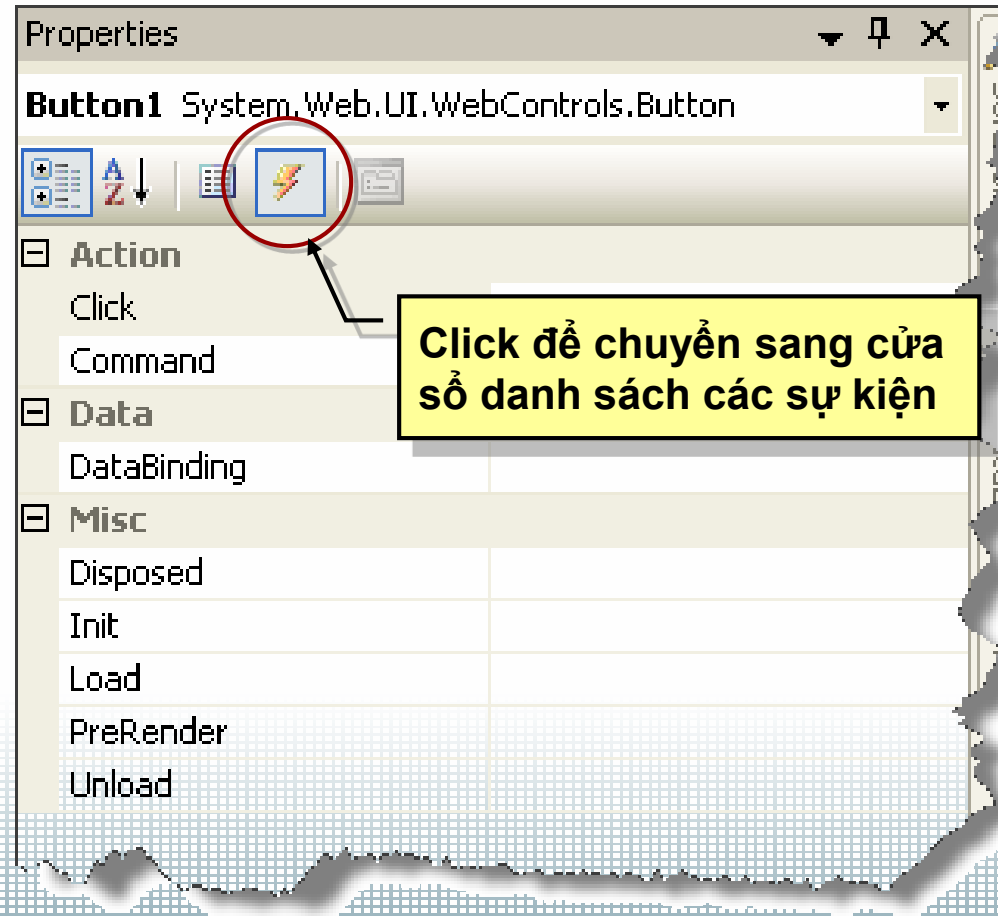
❑ Giới thiệu về sự kiện (Event)

- ❑ Ứng dụng ASP.NET sử dụng mô hình lập trình hướng sự kiện
 - ❑ **Event:** Click, Load, ...
 - ❑ **Event Handler:** Phương thức xử lý khi có sự kiện xảy ra.

```
10
11 public partial class _Default : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16     protected void Button1_Click(object sender, EventArgs e)
17     {
18         this.Label1.Text=this.TextBox1.Text;
19     }
20 }
21
```

Các sự kiện trong ASP.NET (3)

❑ Giới thiệu về sự kiện (Event)



Các sự kiện trong ASP.NET (4)

❑ Page Post-back

- ❑ Khi một sự kiện xảy ra, Web Server sẽ triệu gọi hàm xử lý sự kiện để xử lý; **hàm này được đặt trong chính trang .aspx hoặc từ trang Code-behind.**

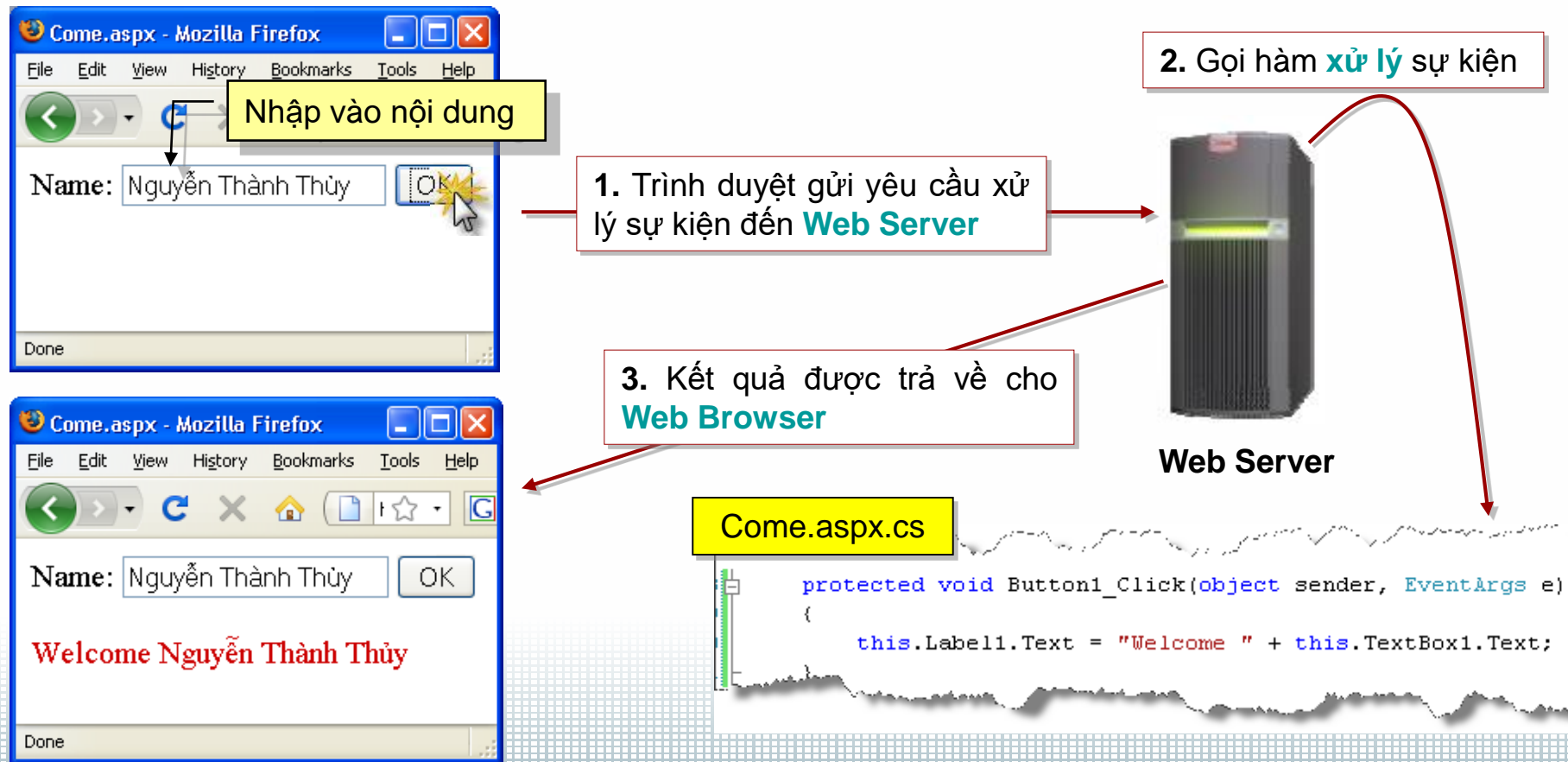
The screenshot displays the Visual Studio IDE with a file named 'Come.aspx'. The 'Client Objects & Events' pane shows '(No Events)'. The code in the editor is as follows:

```
9 <body>
10 <form id="form1" runat="server">
11 <div>
12     Name:
13     <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
14     <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="OK" />
15     <asp:Label ID="Label1" runat="server" ForeColor="#CC0000" Width="239px">
16     </asp:Label></div>
17 </form>
18 </body>
19 </html>
```

A yellow callout box with the text 'Khái báo sự kiện **OnClick**' points to the **OnClick="Button1_Click"** attribute in the button's declaration. Below the code, a preview of the web page is shown in a Mozilla Firefox browser window. The preview displays a form with a label 'Name:', a text input field, and an 'OK' button. The status bar at the bottom of the browser window shows 'Done'.

Các sự kiện trong ASP.NET (5)

❑ Page Post-back

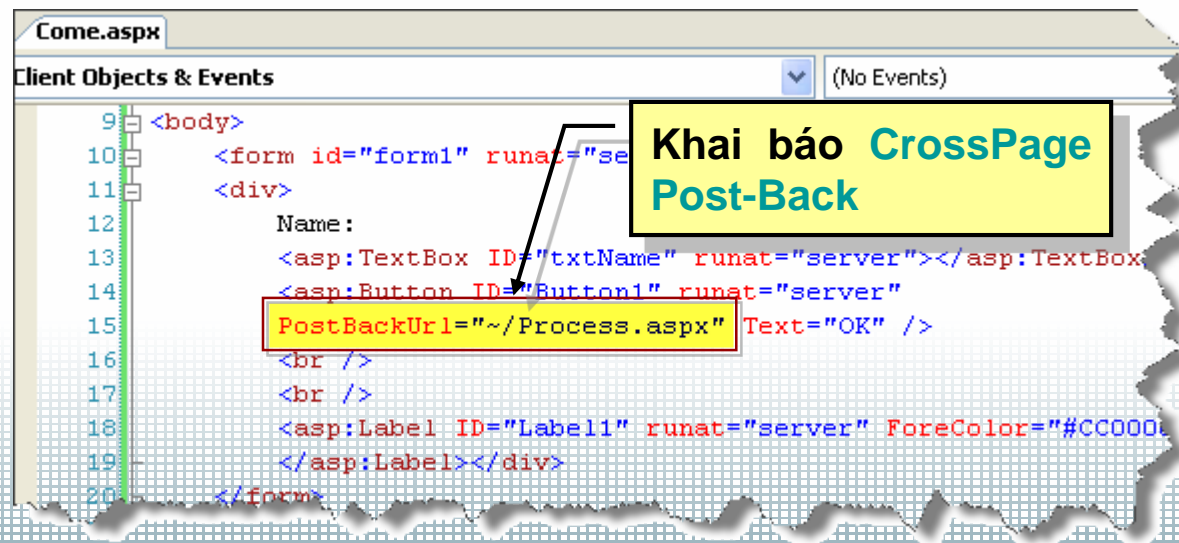
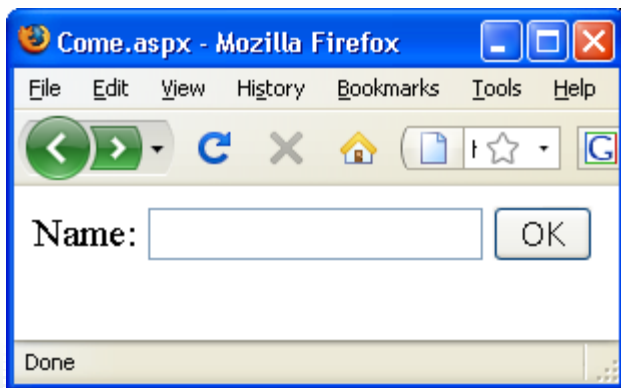


Ví dụ 3.1

Các sự kiện trong ASP.NET (6)

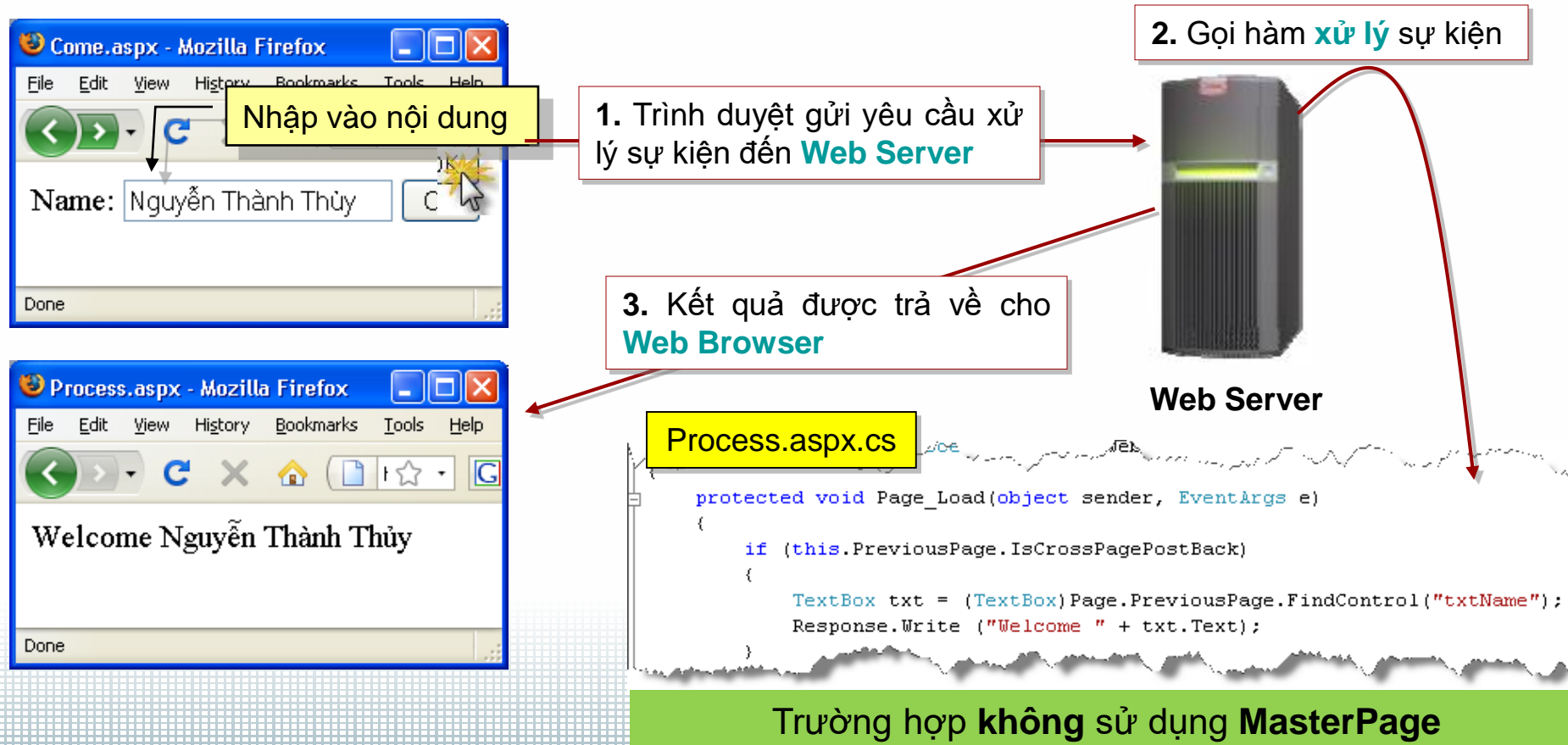
❑ CrossPage Post-back

- ❑ Khi một sự kiện xảy ra, Web Server sẽ triệu gọi hàm xử lý sự kiện để xử lý; **hàm này được đặt trong một trang khác** (không phải trang Code-behind **.aspx.cs** hay trang **.aspx**)



Các sự kiện trong ASP.NET (7)

❑ CrossPage Post-back



Ví dụ 3.2

Các sự kiện trong ASP.NET (7)

❑ CrossPage Post-back

Process.aspx.cs

```
protected void Page_Load(object sender, EventArgs e)
{
    if (this.PreviousPage.IsCrossPagePostBack)
    {
        ContentPlaceHolder cpholder = (ContentPlaceHolder)PreviousPage.Master.FindControl
            ("ContentPlaceHolder1");

        if (cpholder != null)
        {
            TextBox txt = (TextBox)cpholder.FindControl("txt_HoTen");
            if (txt != null)
            {
                this.lbl_ThongBao.Text = txt.Text;
            }
        }
    }
}
```

Trường hợp **có** sử dụng **MasterPage**

Các sự kiện trong ASP.NET (6)

- ❑ **Bài tập 3.3**, thiết kế trang web theo mẫu sau bằng 2 cách: Cách 1. **Page Postback**, Cách 2. **Cross Page Postback**.

Họ tên	<input type="text" value="Nguyễn Thành Thủy"/>
Email	<input type="text" value="thuynt@due.edu.vn"/>
SĐT	<input type="text" value="0905123456"/>
	<input type="button" value="Đăng ký"/>



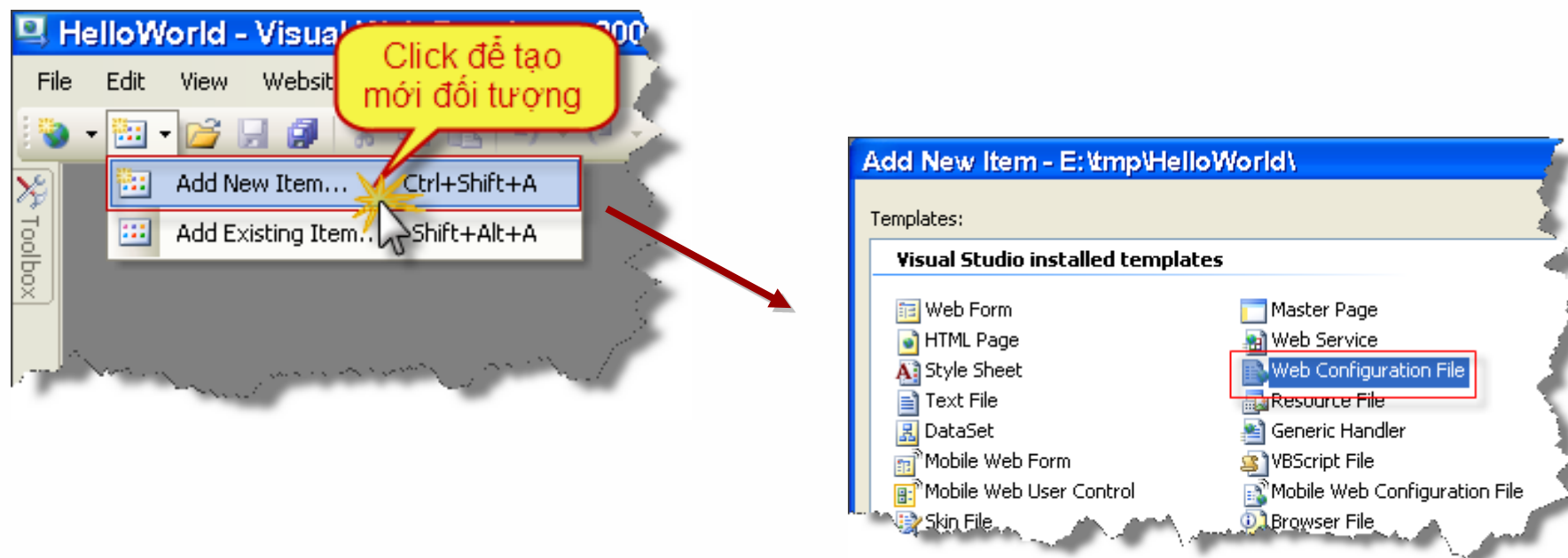
Thông tin đã đăng ký:

- Họ tên: Nguyễn Thành Thủy
- Email: thuynt@due.edu.vn
- SĐT: 0905123456

Cảm ơn quý khách đã đăng ký thông tin!

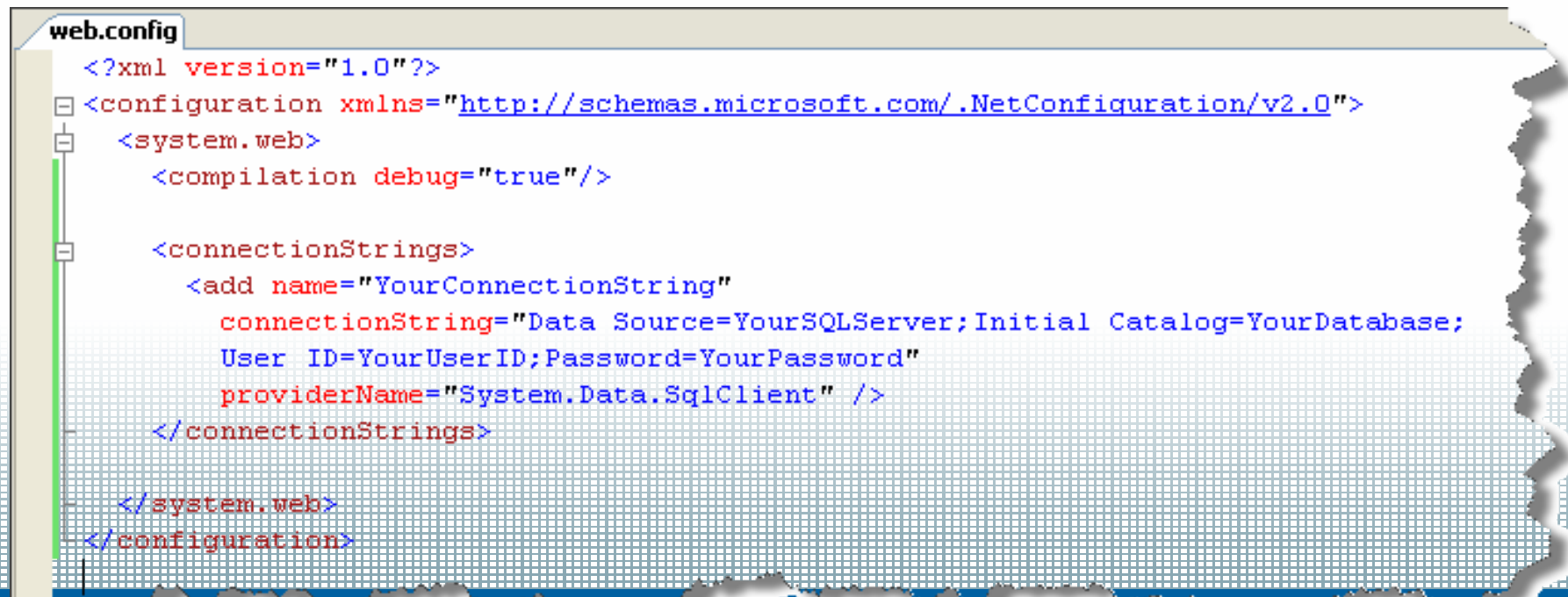
Tập tin web.config

❑ Thêm mới tập tin **Web.config**



Tập tin web.config (2)

- ❑ Tập tin **Web.config** có chức năng gì ?
 - ❑ Là file có định dạng **XML**, chứa các **thông tin cấu hình** cho ứng dụng web.
 - ❑ Mỗi thư mục trong dự án web có thể có 1 file **web.config**. Khi trang **.aspx** được xử lý, hệ thống sẽ triệu gọi trang web.config gần nhất.

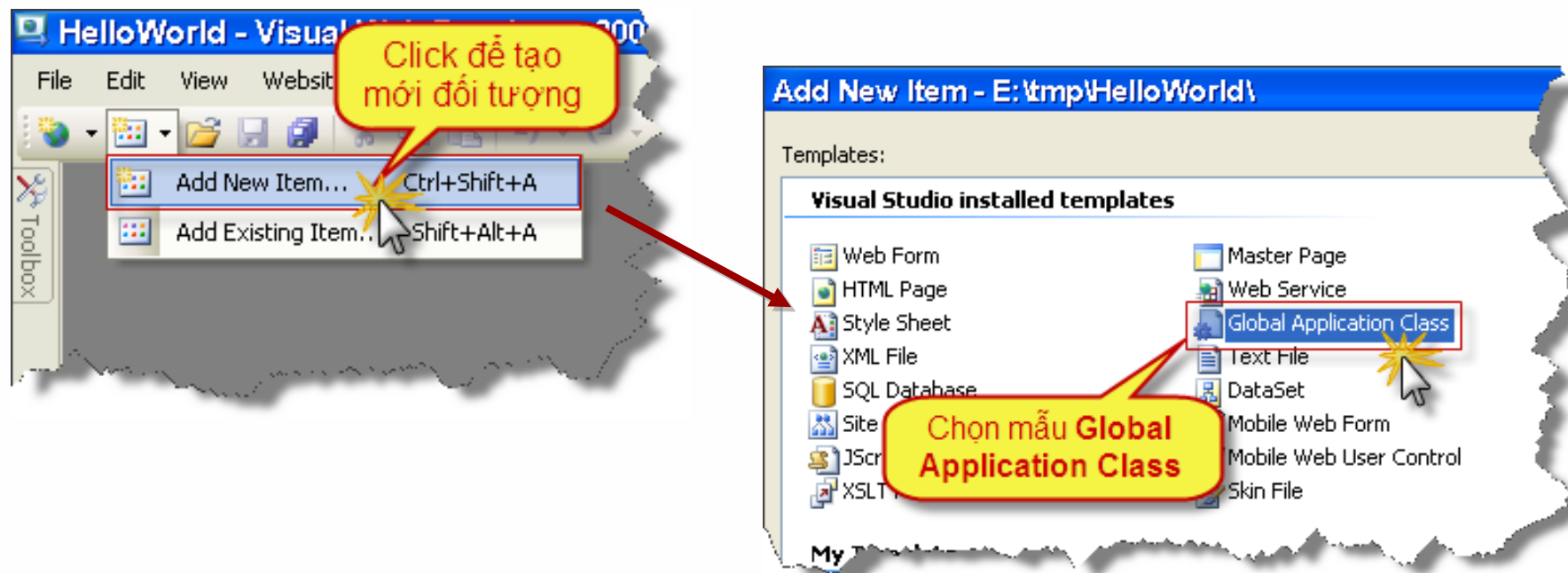
A screenshot of a code editor showing the content of a web.config file. The file is titled 'web.config' in the tab. The code is XML and includes configuration for system.web, connectionStrings, and a specific connection string named 'YourConnectionString'.

```
<?xml version="1.0"?>
<configuration xmlns="http://schemas.microsoft.com/.NetConfiguration/v2.0">
  <system.web>
    <compilation debug="true"/>

    <connectionStrings>
      <add name="YourConnectionString"
        connectionString="Data Source=YourSQLServer;Initial Catalog=YourDatabase;
        User ID=YourUserID;Password=YourPassword"
        providerName="System.Data.SqlClient" />
    </connectionStrings>
  </system.web>
</configuration>
```

Tập tin Global.asax

❑ Thêm mới tập tin Global.asax



Tập tin Global.asax (2)

- ❑ Tập tin **Global.asax** có chức năng gì ?
 - ❑ Cho phép khai báo các **biến cố** để nắm bắt các hành động của người sử dụng web.
 - ❑ Người sử dụng không thể truy cập trực tiếp tập tin Global.asax.

- ❑ Các biến cố trong tập tin **Global.asax**
 - ❑ **Nhóm sự kiện ứng dụng (Application)**
 - ❑ Application_Start
 - ❑ Application_End
 - ❑ Application_Error
 - ❑ **Nhóm sự kiện phiên truy cập (Session)**
 - ❑ Session_Start
 - ❑ Session_End

Tập tin Global.asax (3)

❑ Application_Start

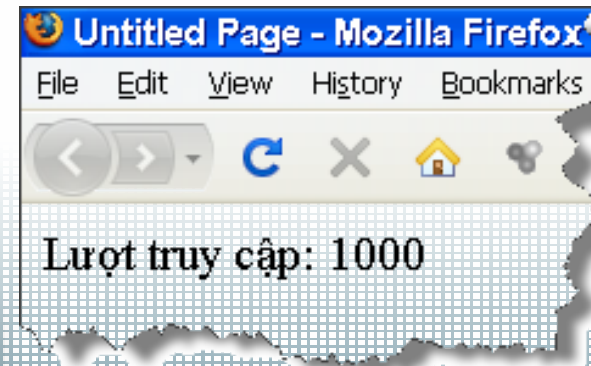
- ❑ Biến cố này sẽ được gọi khi ứng dụng nhận yêu cầu lần đầu tiên.
- ❑ Thường được sử dụng để khởi tạo giá trị ban đầu cho ứng dụng.
 - ❑ Để sử dụng biến cố Application, cần kích hoạt tiện ích **aspnet_state** bằng lệnh: **net start aspnet_state**

Global.asax

```
void Application_Start(object sender, EventArgs e)
{
    Application["LuongTruyCap"] = 1000;
}
```

Default.aspx.cs

```
protected void Page_Load(object sender, EventArgs e)
{
    this.lbl_TongLuotTruyCap.Text = (Application["LuongTruyCap"].ToString());
}
```



Tập tin Global.asax (4)

❑ **Application_End**

- ❑ Biến cố này sẽ được gọi khi ứng dụng kết thúc.
- ❑ Thường được sử dụng để giải phóng tài nguyên hoặc ghi dữ liệu của ứng dụng vào database trước khi ứng dụng kết thúc.

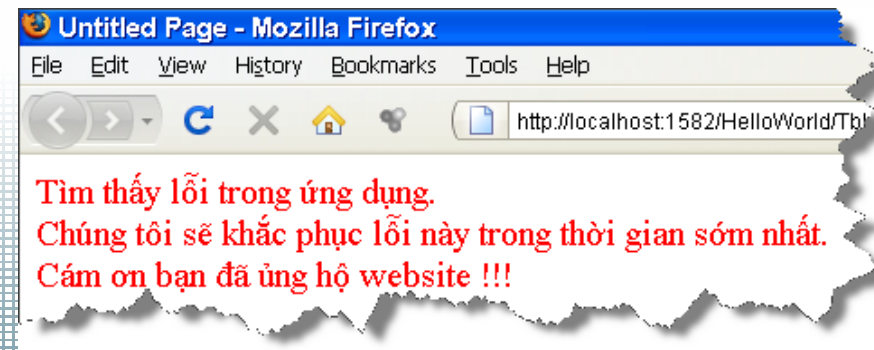
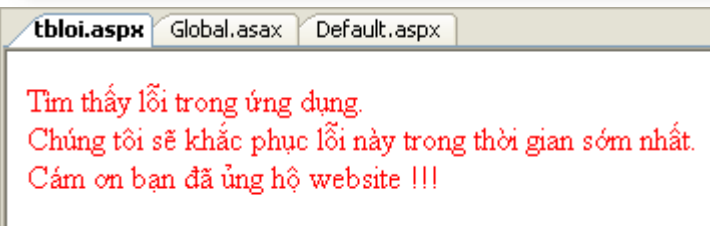
Tập tin Global.asax (5)

❑ Application_Error

- ❑ Biến cố này sẽ được gọi khi xuất hiện lỗi ở bất kỳ trang web nào trong ứng dụng.
- ❑ Thường được sử dụng để kiểm soát lỗi của ứng dụng.

Global.asax

```
void Application_Error(object sender, EventArgs e)
{
    Response.Redirect("TbLoi.aspx");
}
```



Tập tin Global.asax (6)

❑ Session_Start

- ❑ Biến cố này sẽ được gọi khi có một phiên làm việc mới được tạo ra.
- ❑ Thường được sử dụng để kiểm soát việc truy cập của người sử dụng, như: đăng nhập hệ thống, thống kê số lượt truy cập, ...

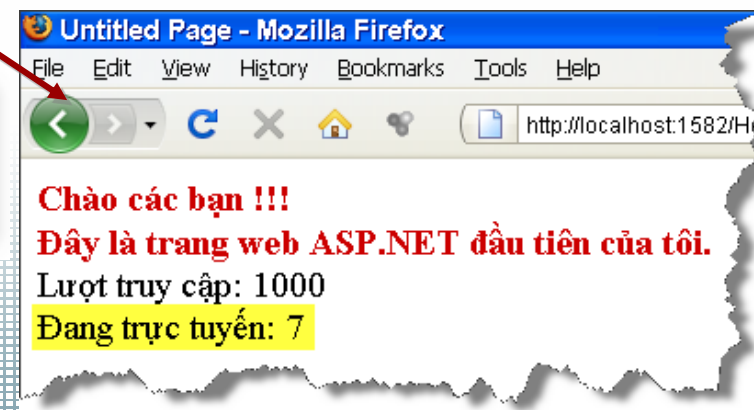
Global.asax

```
void Session_Start(object sender, EventArgs e)
{
    Application["TrucTuyen"] = (int)Application["TrucTuyen"] + 1;
}
```

Default.aspx

Lượt truy cập: <%=Application["LuongTruyCap"] %>

Đang trực tuyến: <%=Application["TrucTuyen"] %>



Tập tin Global.asax (7)

❑ Session_End

- ❑ Biến cố này sẽ được gọi khi có một phiên làm việc được kết thúc.

Global.asax

```
void Session_End(object sender, EventArgs e)
{
    Application["TrucTuyen"] = (int)Application["TrucTuyen"] - 1;
}
```

Tập tin Global.asax (7)

- ❑ **Bài tập 3.4**, thiết kế trang web thực hiện đếm lượt truy cập và lượt online website theo mẫu sau.

Số lượt truy cập website: **4**

Số lượt Online: **1**

- ❑ **Bài tập 3.5**, thiết kế trang thông báo lỗi theo mẫu sau.

Hệ thống tìm thấy lỗi khi thực hiện,
chúng tôi sẽ khắc phục lỗi này trong thời gian sớm nhất.
Xin cảm ơn!!!

Tập tin Global.asax (7)

- ❑ **Bài tập 3.6**, thiết kế trang web sử dụng biến session để kiểm soát đăng nhập hệ thống.
- Trang **login.aspx**: cho phép người dùng đăng nhập hệ thống. Đăng nhập sai thì thông báo “**Đăng nhập không thành công!!!**”. Nếu đăng nhập đúng (admin/admin) chuyển người dùng vào trang **quantri.aspx**
- Ở trang **quantri.aspx**: nếu người dùng chưa login thì tự động redirect về trang **login.aspx**

