



# Chương 8

---

## **Tùy biến điều khiển trong ASP.NET**



# Mục đích

---

- *Xác định cần thiết của việc tạo custom controls*
- *Tạo điều khiển đơn giản dùng ASP.NET*
- *Tạo Composite control dùng C#*
- *Sử dụng sự kiện với custom controls*



# Controls

---

- Các lớp được xây dựng cho việc tái sử dụng mã  
→ cần thiết tái sử dụng lại các điều khiển giao diện người dùng
- Control giúp tái sử dụng một cách trực quan cũng như bổ sung các chức năng tương ứng
- Giúp đóng gói các khả năng và phân phối
- ASP.NET controls dùng cho Web để tạo HTML hoặc WML, hiển thị trên trình duyệt người dùng

# Controls ...

Assfffgghfiol  
hjkklkjklkjklkj  
Uyuiiyui  
ghghghjjs



DataGrid

Code for a datagrid

Assfffgghfiol  
hjkklkjklkjklkj  
iyuiiyui  
ghghghjjs



DataView

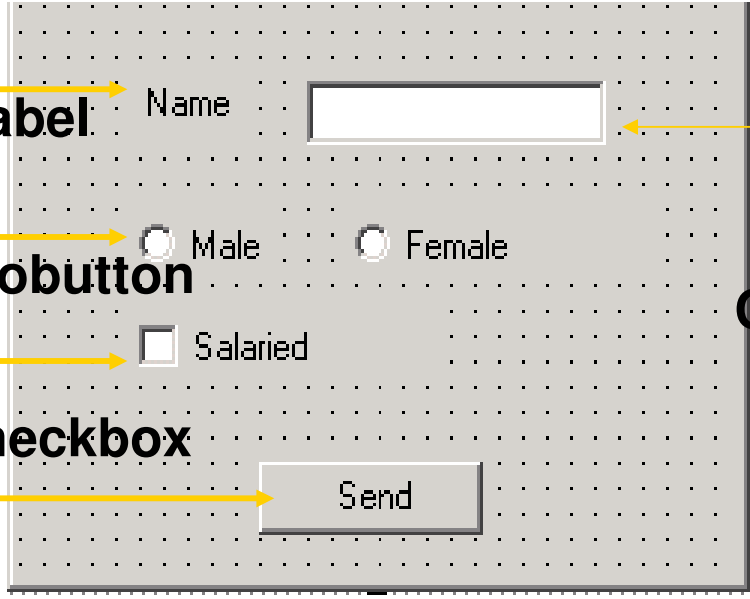
Code for a dataview

Assfffgghfiol  
hjkklkjklkjklkj  
Uyuiiyui  
ghgh



Label

Code for a Label



abl TextBox

Code for a textbox

Assfffgghfiol  
hjkklkjklkjklkj  
Uyuiiyui  
gh



RadioButton

Code for a radiobutton

Assfffgghfiol  
hjkklkjklkjklkj  
Uyuiiyui  
gh



CheckBox

Code for a checkbox

ab Button

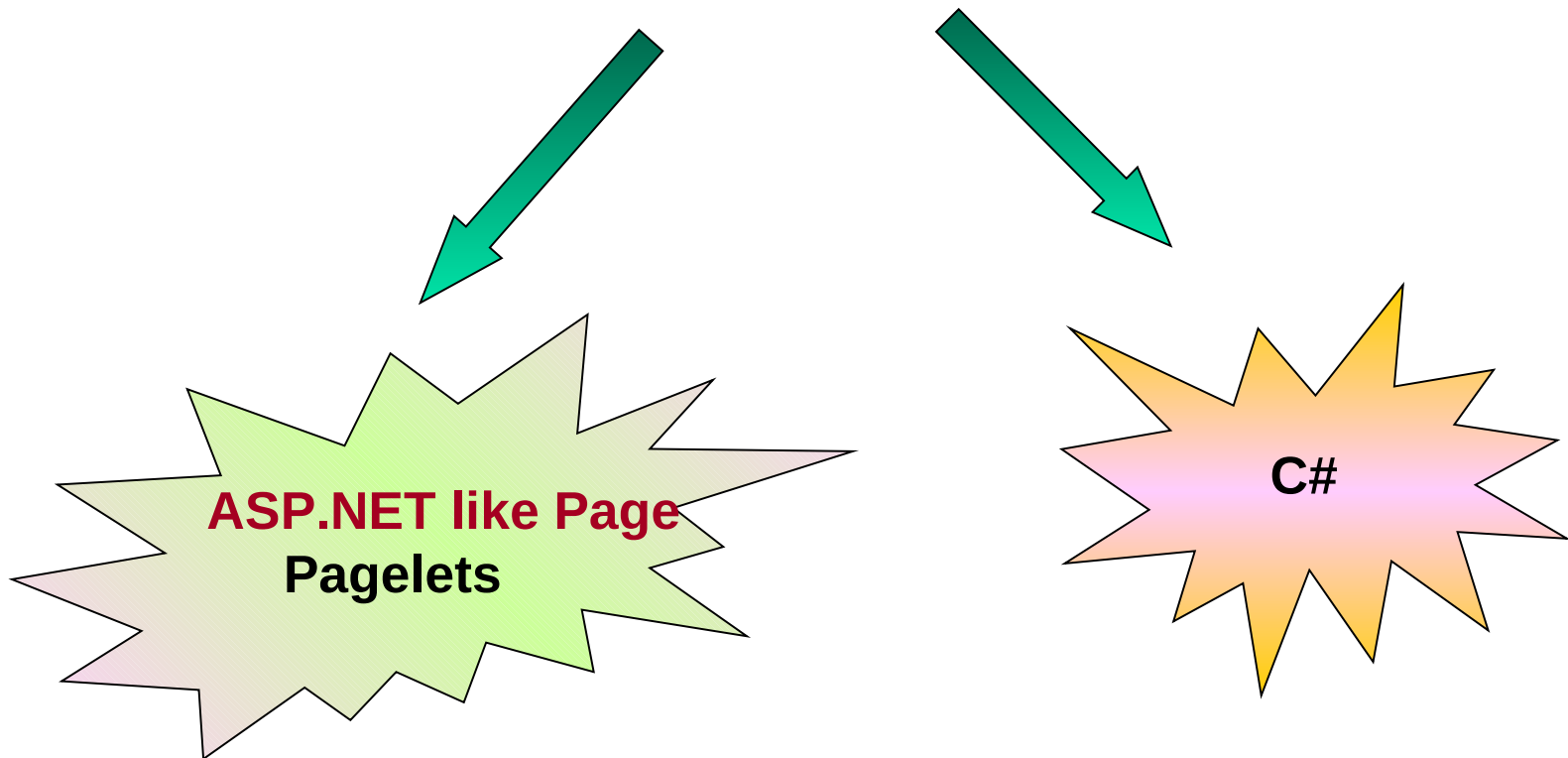
Code for a button



# Custom Controls

---

**Tạo theo 2 cách**



# Custom Control sử dụng Label Control

**MyLabel.ascx**

```
<% @Control Language="C#" Debug="True" %>
<ASP:Label id="lblOne" runat="server" style="color:Red;
font-family: arial; font-size=40pt"></asp:Label>
<script language="C#" runat="server">
public string MyText
{
    get
    {
        // Do Nothing
        return(lblOne.Text);
    }
    set
    {
        lblOne.Text = value;
    }
}
```

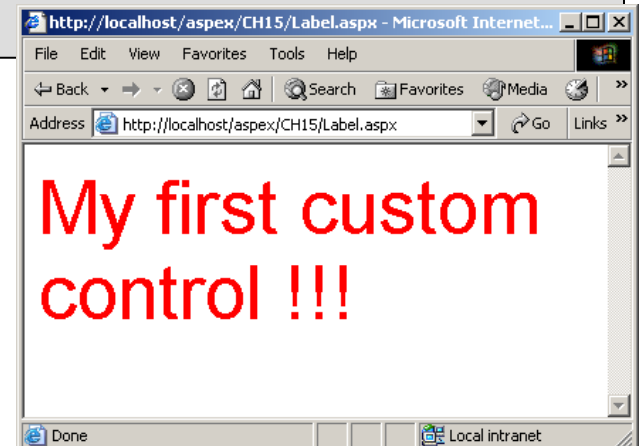
# Custom Control dùng Label Control

```
}  
</script>
```

## Label.aspx

```
<% @Register Tagprefix="MyFirstControl" TagName="MyLb1"  
Src="MyLabel.ascx" %>
```

```
<MyFirstControl:MyLb1 runat="Server" id = "AllMine"  
MyText="My first custom control !!!" />
```





# Custom Controls dùng C#

---

Điều khiển này được viết hoàn toàn dùng C# mà không cần dùng bất cứ điều khiển ASP.Net nào

## **Repeater.cs**

```
using System;
using System.Web;
using System.Web.UI;
namespace MyOwnControls
{
    public class TextRepeater : Control
    {
        protected override void Render(HtmlTextWriter writer)
        {
            int intCount;
```





# Custom Controls dùng C#

---

```
for (intCount=1; intCount<=3; intCount++)
{
    writer.Write ("<h1>This is a custom control</h1>
<br>");
}
}
}
}
```

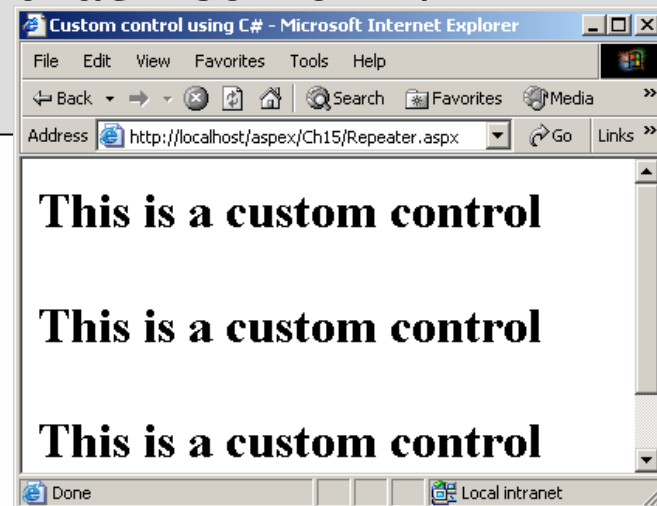
Mã được lưu trong tập tin cs và được biên dịch thành dll và nên được đặt vào trong thư mục bin của ứng dụng

```
csc /t:library /r:System.dll, System.Web.Dll Repeater.cs
```

# Custom Controls dùng C#

Sử dụng trong rang ASP.Net

```
<% @Register TagPrefix="Mine" Namespace="MyOwnControls"
Assembly="Repeater" %>
<html>
<title>Custom control using C#</title>
  <body>
    <Mine:TextRepeater runat="server" />
  </body>
</html>
```





# Thuộc tính của Custom Controls

---

- Để cho phép sử dụng tốt hơn các control, chúng ta có thể tạo thuộc tính cho các điều khiển
- Các thuộc tính này có thể thay đổi một cách tự động

## NewRepeater. cs

```
using System;
using System.Web;
using System.Web.UI;
namespace MyOwnControls
{
    public class TextRepeater : Control
    {
        public int timesToRepeat;
```



# Properties of Custom Controls

---

```
public string myText;
public int TimesToRepeat
{
    get
    {
        return timesToRepeat;
    }
    set
    {
        if (timesToRepeat > 10)
            throw new ArgumentException ("The text should
not be repeated more than 10 times");
        else
            timesToRepeat = value;
    }
}
```



# Properties of Custom Controls

```
public string MyText
{
    get
    {
        return myText;
    }
    set
    {
        if (myText.Length > 25)
            throw new ArgumentException("The text
should not be more than 25 characters");
        else
            myText = value;
    }
}
```



# Properties of Custom Controls

---

```
protected override void Render(HtmlTextWriter writer)
{
    for (int Count = 1; Count <= TimesToRepeat; Count++)
    {
        writer.Write("<h1>" + MyText + "</h1><br>");
    }
}
}
```



# Composite Controls

---

- Các control đơn giản có thể kết hợp với nhau để tạo các control phức tạp → **Composite controls**.
- Composite controls có thể bao hàm các custom controls, cũng như các control của windows
- Mỗi trang asp.net bao gồm ít nhất một control → là một composite control



# Composite Controls – Ví dụ

---

## Composite.cs

```
using System;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
namespace MyOwnControls
{
    public class MyComposition : Control, INamingContainer
    {
        public int Val
        {
            get
            {
                this.EnsureChildControls();
                return Int32.Parse (((TextBox) Controls[1]).
Text);
            }
        }
    }
}
```





# Composite Controls - Ví dụ

---

```
    set
    {
        this.EnsureChildControls();
        ((TextBox)Controls[1]).Text = value.ToString();
    }
}
protected override void CreateChildControls()
{
    this.Controls.Add(new LiteralControl("<h3>Value: "));
    TextBox box = new TextBox();
    box.Text = "0";
    this.Controls.Add(box);
    this.Controls.Add(new LiteralControl("</h3>"));
}
}
}
```



# Composite Controls – Ví dụ

---

composite control được tạo và sử dụng giống như các control khác trong asp.net

## Composite.aspx

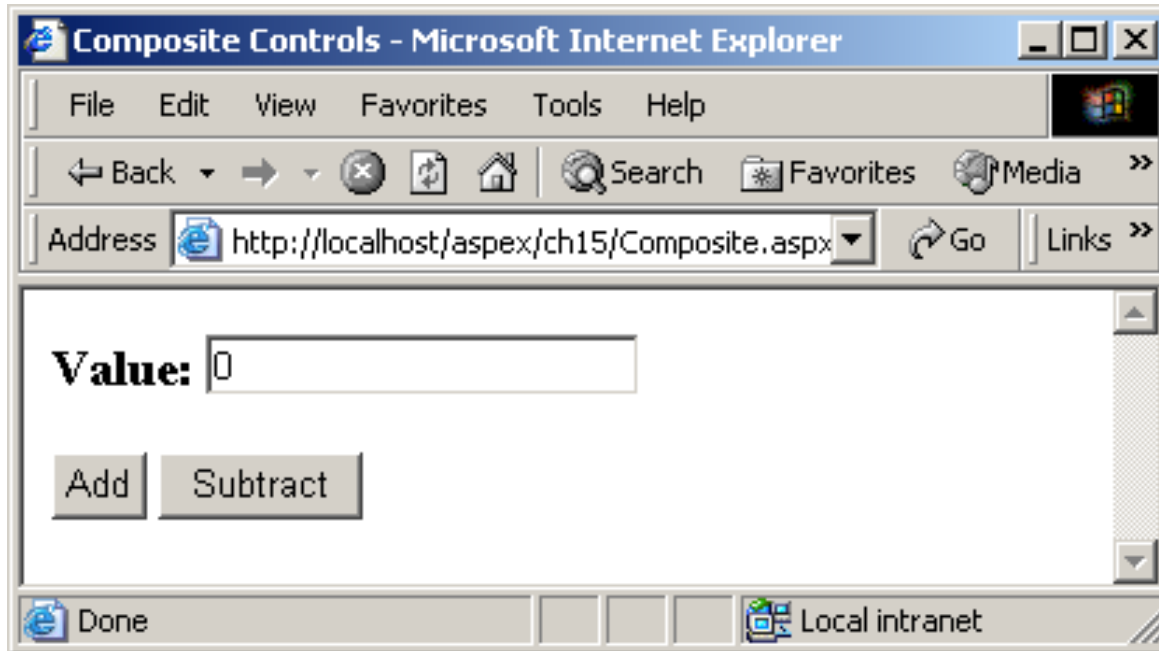
```
<% @Register TagPrefix="MyOwnControls"
Namespace="MyOwnControls" Assembly="Composite" %>
<HTML>
  <title>Composite Controls</title>
  <script language="C#" runat="server">
    private void AddBtn_Click(Object sender, EventArgs e)
    {
      MyComposition1.Val = MyComposition1.Val + 1;
    }
    private void SubtractBtn_Click(Object sender, EventArgs
e)
```



# Composite Controls – Ví dụ

```
{
    MyComposition1.Val = MyComposition1.Val - 1;
}
</script>
<body>
    <form method="post" action="ThisPage.aspx" runat=
"server" ID="Form1">
        <MyOwnControls:MyComposition id="MyComposition1"
runat="server" />
        <asp:button text="Add" OnClick="AddBtn_Click"
runat="server" ID="btnAdd" />
        <asp:button text="Subtract" OnClick="SubtractBtn_
Click" runat="server" ID="btnSubtract" />
    </form>
</body>
</HTML>
```

# Composite Controls – Ví dụ





# Summary

---

- Controls giúp tái sử dụng mã cũng như chức năng của nó
- Custom web controls được tạo theo 2 cách:
  - Pagelets
  - Web Controls dùng C#
- Pagelets là custom control nhưng giống một trang asp, và có phần mở rộng là .ascx.
- Web Controls render HTML tự động.
- Custom Controls là kết hợp của các control khác
- Giao diện **System.Web.UI.INamingContainer** không có phương thức, ASP.Net dùng để tạo các unique ID.