

# Chapter 3

## Working with the Standard Web Server Controls

PRENTICE  
HALL

### core

# INTERNET

APPLICATION DEVELOPMENT  
WITH **ASP.NET 2.0**



▼ Get up to speed quickly with numerous real-world walkthrough exercises, code listings, in-depth examples, and snippets

▼ Learn valuable design principles and best practices of ASP.NET Web application development

▼ Gain in-depth knowledge of many key ASP.NET 2.0 topics, including controls, data binding, security, Web services, and a sneak peek at ASP.NET AJAX

RANDY CONNOLLY

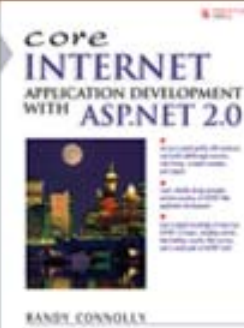
---

# Overview

- This presentation covers the initial overview of the different server controls.

## 2 Essential Server Controls

COPYRIGHT © 2007 RANDY CONNOLLY



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

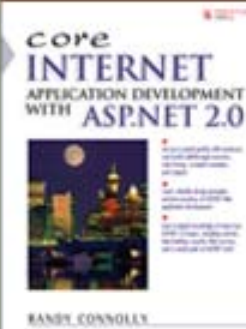
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Introducing Server Controls

- Normal HTML tags such as `<input>`, `<H1>` and `<select>` are not processed by the server but are sent to and displayed by the browser.
- Server controls, in contrast, are tags that can be understood by the server.
- Each ASP.NET server control has an object model containing properties, methods, and events.

## 3 Essential Server Controls

COPYRIGHT © 2007 RANDY CONNOLLY

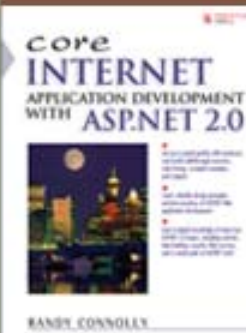


CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Introducing Server Controls

- There are five kinds of server controls:
  - HTML server controls
  - Web server controls
  - Validation server controls
  - User controls
  - Custom server controls
- All five of these different types of controls can be used within a single given Web form.



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

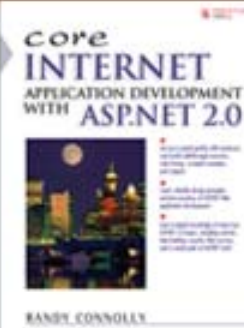
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# HTML Server Controls

- Most standard HTML tags can be turned into HTML server controls simply by adding the `runat="server"` attribute.
  - This `runat` attribute indicates that the element is to be processed on the server.
  - You can this programmatically respond to events or bind data.

## 5 Essential Server Controls

COPYRIGHT © 2007 RANDY CONNOLLY



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

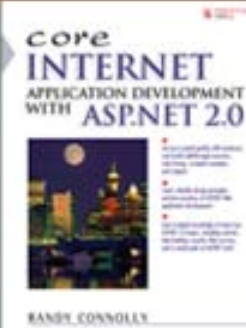
Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# HTML Server Controls

- HTML server controls can be useful for:
  - situations in which you need complete control over how the HTML element will be rendered in the browser, or
  - when migrating an existing ASP page to ASP.NET.
- Web server controls are almost always preferable to HTML server controls due to their richly typed object model.

## 6 Essential Server Controls

COPYRIGHT © 2007 RANDY CONNOLLY



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

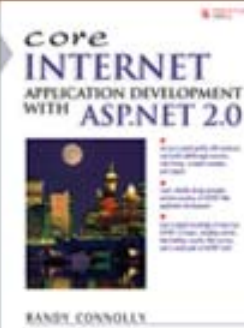
# Web Server Controls

- Web server controls are also created on the server and they require a `runat="server"` attribute to work.

```
<asp:Button ID="myButton" runat="server" />
```

## 7 Essential Server Controls

COPYRIGHT © 2007 RANDY CONNOLLY



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

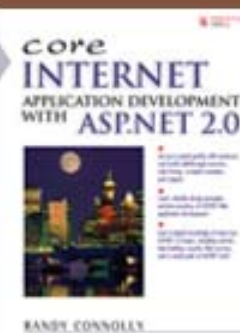
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Web Server Controls

- Some Web server controls represent traditional HTML form elements such as buttons and drop-down lists.
- Other Web server controls represent more complex or abstract elements such as calendars, data lists, data sources, and tree views.
  - These more complex Web server controls do not necessarily map one-to-one (or even at all) to any existing HTML tags and can in fact be realized by dozens if not hundreds of HTML tags and many lines of Javascript code.

## 8 Essential Server Controls

COPYRIGHT © 2007 RANDY CONNOLLY



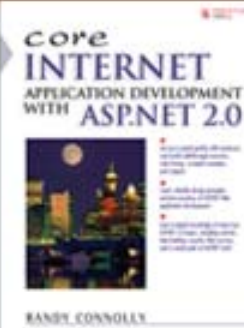
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Validation Controls

- These controls allow you to test a user's input for validity.
  - They are actually a special type of Web server control.
- Validation controls encapsulate common user input validation checks required by most Web applications:
  - ensuring that a required field is not empty,
  - comparing an input value against another value,
  - checking if a value falls within a range,
  - verifying that an input value matches a given pattern.
- Covered in Chapter 5.



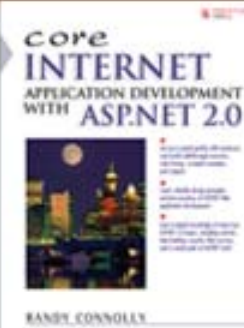
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# User Controls

- These are developer-created controls that use the same programming techniques used to write Web forms pages.
- They typically allow the encapsulation of the functionality of multiple server controls along with other ASP.NET or HTML content in a single unit.
- Covered in Chapter 6.



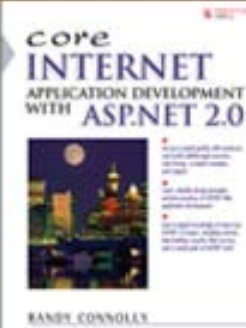
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Custom Server Controls

- A custom server control is a Web server control that you can create. A custom control is a compiled class and may combine multiple existing server controls.
- A custom control, unlike a user control, contains no declarative elements, and can be extended, use templates, support data binding, and be redistributed in a precompiled assembly.
- Partially covered in Chapter 14.



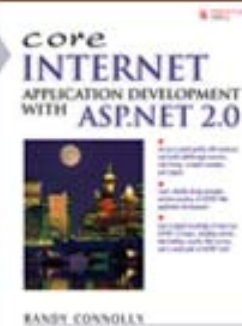
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Web Server Control Overview

- Web server controls are added to a Web form in the same way as any HTML element:
  - You can type the markup code in Source view in VS.
  - You can use drag-and-drop from Source or Design view.
  - As well, you can programmatically add controls at runtime.



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

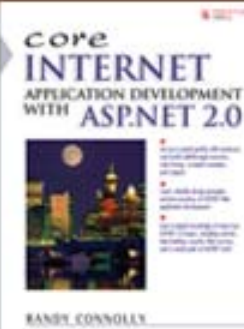
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Web Server Control Syntax

```
<asp:controlName id="some_id" runat="server"
  «other attributes» >
</asp:controlName>
```

## 13 Essential Server Controls

COPYRIGHT © 2007 RANDY CONNOLLY

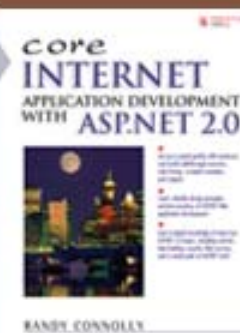


CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# List of the Common Web Server Controls

BulletedList	Displays a bulleted list of items.
Button	Displays a push button that posts a Web form page back to the server.
Calendar	Displays a month calendar from which the user can select dates.
CheckBox	Displays a check box for selected true or false values.
CheckBoxList	Displays a multiselection check box group.
DropDownList	Displays a drop-down list for selecting a value from a list of values.



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# List of the Common Web Server Controls

HiddenField	Stores a nondisplayed value in a form that needs to be persisted across posts.
HyperLink	Displays a hyperlink that when clicked requests a different page.
Image	Displays an image.
ImageButton	Displays an image that posts the form back to the server
ImageMap	Displays an image with predefined hot spot regions that post back to the server or navigate to a different page.
Label	Displays static content that can be set programmatically and whose content can be styled.



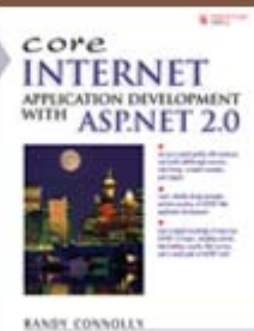
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# List of the Common Web Server Controls

LinkButton	Creates a hyperlink-style button that posts the form back to the server.
ListBox	Displays a hyperlink that when clicked requests a different page. Creates a single- or multiselection list.
Literal	Like the Label, displays static content that is programmable. Unlike the Label control, it does not let you apply styles to its content.
RadioButton	Creates a radio button form element.
RadioButtonList	Creates a group of radio button form elements.
Table	Creates an HTML table; principally used for programmatically constructing a table.
TextBox	Creates a text box form element.

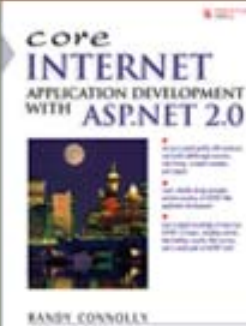


CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

## Common Members

- Learning how to work with all of these controls might seem a daunting prospect.
- Thankfully, Web server controls share a common object model.



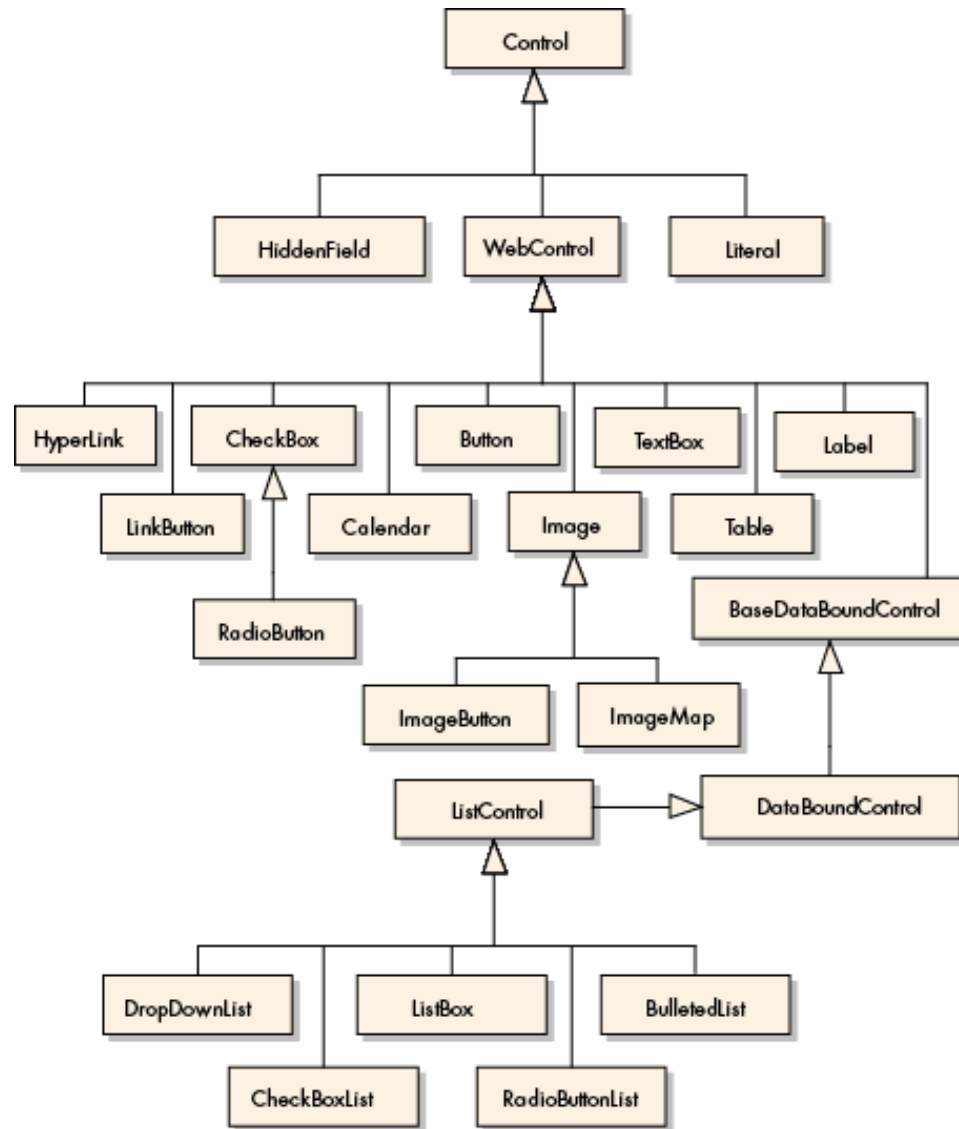
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Object Model

## 18 Essential Server Controls

COPYRIGHT © 2007 RANDY CONNOLLY



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

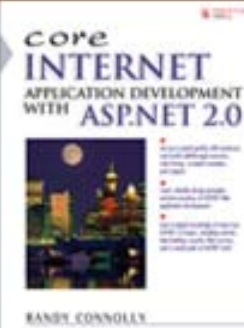
Core  
INTERNET  
APPLICATION DEVELOPMENT  
WITH ASP.NET 2.0



RANDY CONNOLLY

## Common Members

- Most Web server controls inherit from the `WebControl` class.
  - This `WebControl` class in turn inherits from the `Control` class.
- Both of these base classes define a variety of properties, methods, and events that are available to all controls derived from them.
  - Most of these modify the formatting and display of the controls.



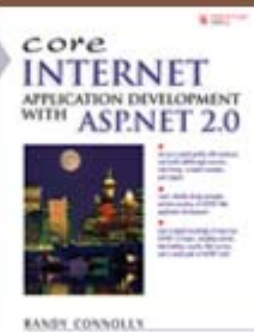
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

## Some Properties of the WebControl Class

BackColor	The background color (either standard HTML color identifier or the name of the color) of the control.
BorderWidth	The thickness (in pixels) of the control's border.
CssClass	The CSS class name assigned to the control.
Font	Font information for the control. This property contains <b>subproperties</b> .
ToolTip	The tool tip text (i.e., the text that appears when the mouse rests over control) for the control.
Width	The width of the control.



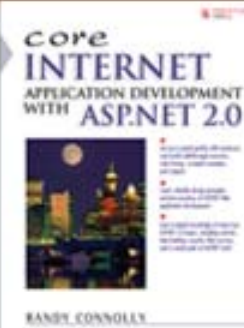
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

## Some Properties of the Control Class

Id	The unique identifier for the control.
Visible	Specifies whether the control is visible.



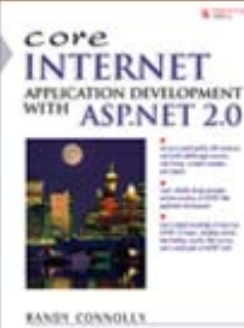
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Subproperties

- Properties can also have properties;
  - these are called **subproperties**.
- For instance, the `Font` property is a complex object with properties of its own, such as `Name` and `Size`.
- When working with subproperties programmatically, you use dot notation.
  - e.g.,  
`somecontrol.Font.Size=10;`
- When working with subproperties declaratively, you use hyphen (-) notation.
  - e.g.,  
`<asp:Label id="acontrol" Font-Size="10"  
runat="server" />`



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

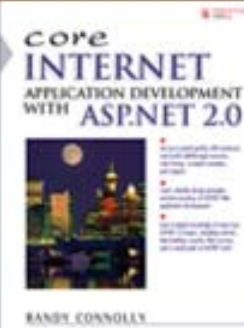
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Manipulating Properties Programmatically

- You can retrieve the value of a property or set the value of a property at runtime.
- These properties are strongly typed and vary depending upon the property.
  - Some properties have a primitive data type such as a Boolean or a numeric.
  - Other property values are defined by an enumerated type or some other type.

```
// Using a primitive
myLabel.Text = "Randy";
string abc = myTextBox.Text;
myLabel.Visible = false;

// Using an enumeration
myBulletedList.BulletStyle = BulletStyle.Circle;
TextBoxMode mode = myTextBox.TextMode;
```



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

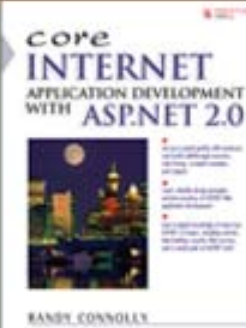
Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Event Properties

- All controls support events.
- You can specify the event handler method for a given event declaratively by affixing the `on` prefix to the event property name.
  - E.g., if you have an event handling method named `btnOne_Click` that you want to run when the user clicks a button, you would use:

```
<asp:button id="btnOne" runat="server"  
    OnClick="btnOne_Click" />
```



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

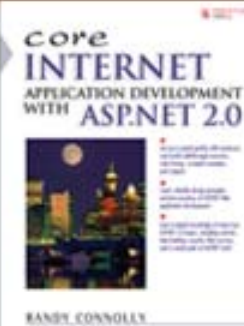
Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Event Methods

- All event methods in the .NET Framework are `void` methods with two parameters:
  - the source object that raised the event
  - the data for the event, usually contained within an `EventArgs` object (or an object derived from it).
- Thus, the method signature for the event handler would be:

```
public void btnOne_Click(object source,  
    EventArgs e)  
{  
    // code goes here  
}
```

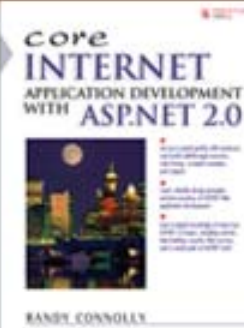


CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

## Unit-Based Measurement Properties

- Various measurement properties such as `Width`, `Height`, and `FontSize` are implemented using the `Unit` structure.
- This `Unit` structure allows you to use any HTML- or CSS-compatible size unit, such as `cm`, `inch`, and `pixels`.
- These size units are defined in the `UnitType` enumeration.

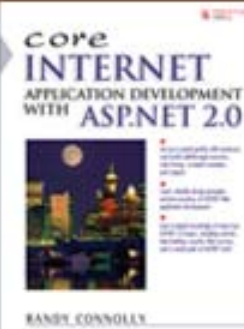


CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Color-Based Properties

- All color-based properties use the `Color` structure.
- The `Color` structure defines:
  - all the HTML 4.0 system-defined colors
  - the color names supported by most browsers
  - methods for retrieving and specifying color using different color models (such as HSB and RGB).
- 



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Collection Properties

- Some properties are not a single structure, primitive, or enumerated type, but a collection of other objects.
- An example of such a property is the `Items` collection of the `DropDownList` control.
  - This collection contains zero or more `Listitem` objects.
  - These collection properties have their own methods and properties for determining the size of the collection, as well for adding, retrieving, and removing items from the collection.

```
DropDownList drpSample = new DropDownList();  
  
// Create the item, then add to collection  
ListItem li = new ListItem("Item 2");  
drpSample.Items.Add(li);  
  
// Combine the creation and addition to collection steps  
drpSample.Items.Add(new ListItem("Item 1"));
```



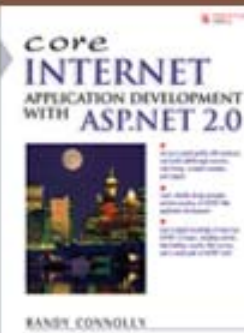
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Additional Control Attributes

- Any given control can be given additional **HTML attributes**.
  - Attributes are name-value pairs that will be added to the rendered HTML.
  - Generally used to provide additional client-side scripting behavior to a control.



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

```
myButton.Attributes["onmouseout"] = "document.bgColor='green'";  
myButton.Attributes["onmouseover"] = "document.bgColor='blue'";
```

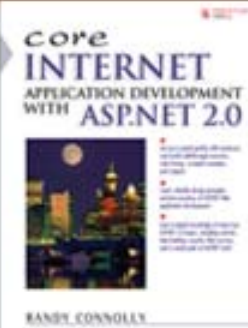
rendered as



```
<input type="submit" name="myButton" value="Click Me" id="myButton"  
  onmouseout="document.bgColor='green'";  
  onmouseover="document.bgColor='blue'"; />
```

# Essential Controls

- See pages 107-176 for details.



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Literal vs Label Control

31 Essential Server Controls

COPYRIGHT © 2007 RANDY CONNOLLY

CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

Core  
INTERNET  
APPLICATION DEVELOPMENT  
WITH ASP.NET 2.0



RANDY CONNOLLY

Label controls are (by default) rendered in the browser as static text within an HTML `<span>` element, e.g.,

```
<asp:Label id="labMsg" runat="server" text="hello"/>
```

will be rendered as

```
<span id="labMsg">hello</span>
```

The `Literal` control does not add any HTML elements to the text, e.g.,

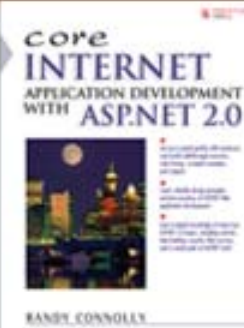
```
<asp:Literal id="litMsg" runat="server" text="hello"/>
```

will be rendered as

```
hello
```

## asp:label as HTML <label>

- You can change the rendering of a `Label` control via the `AccessKey` attribute.
  - It allows the user to use a combination of keyboard keys (on Windows machines, usually the ALT key plus some other key) to move the focus to an associated input control.



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# asp:label as HTML <label>

```
<asp:Label ID="labName" runat="server"
  AccessKey="N"
  AssociatedControlID="txtName"
  Text="<u>N</u>ame" />

<asp:TextBox ID="txtName" runat="server" />
```

rendered as

```
<label for="txtName" id="labName" AccessKey="N" >
  <u>N</u>ame
</label>
<input type="text" id="txtName" />
```

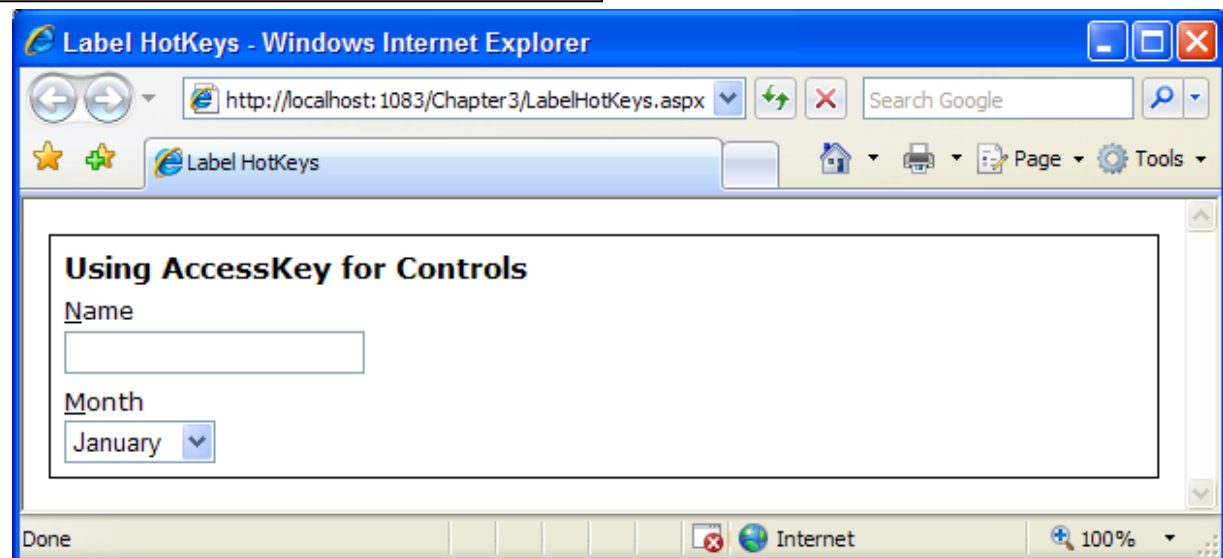
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

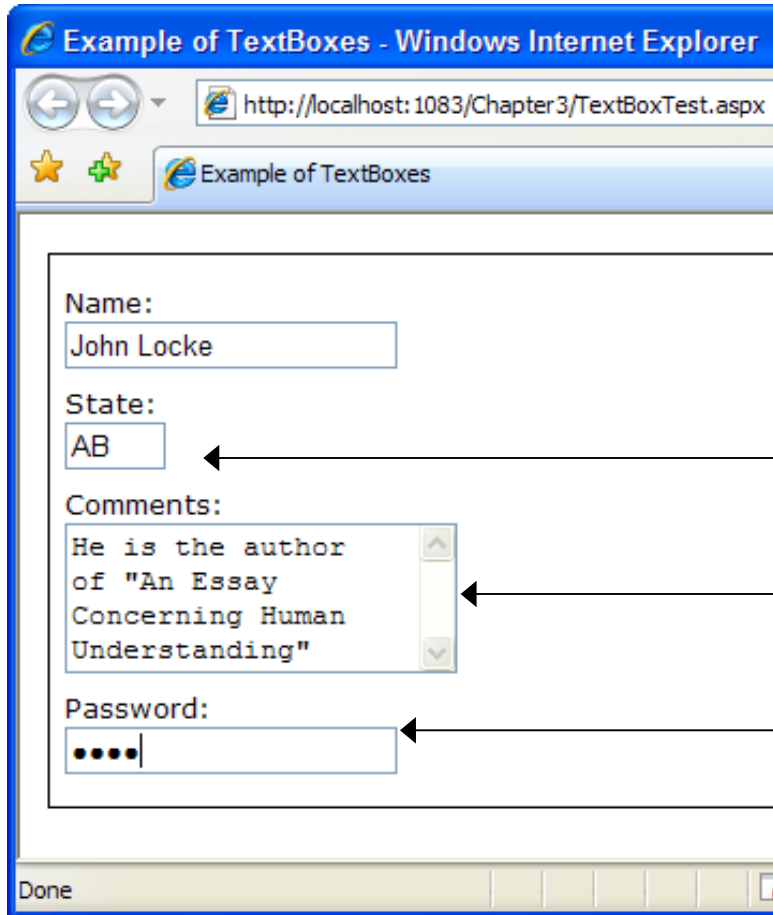
core  
INTERNET  
APPLICATION DEVELOPMENT  
WITH ASP.NET 2.0



RANDY CONNOLLY



# TextBox



← **MaxLength="2"**

← **TextMode="MultiLine"**

← **TextMode="Password"**

CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

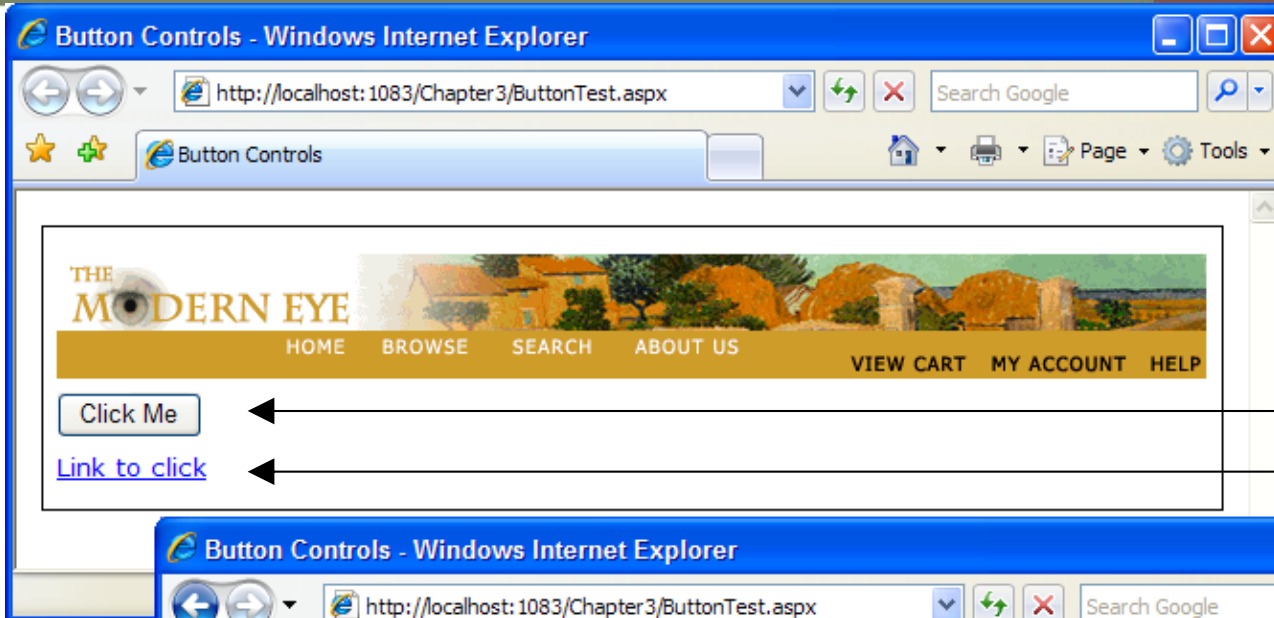
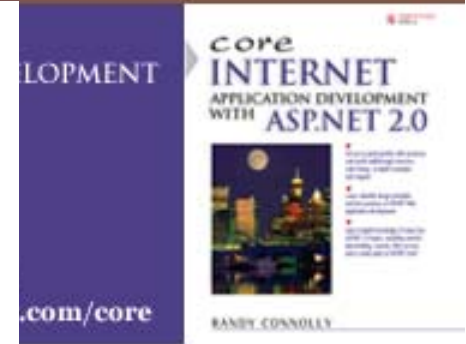
Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

Core  
INTERNET  
APPLICATION DEVELOPMENT  
WITH ASP.NET 2.0



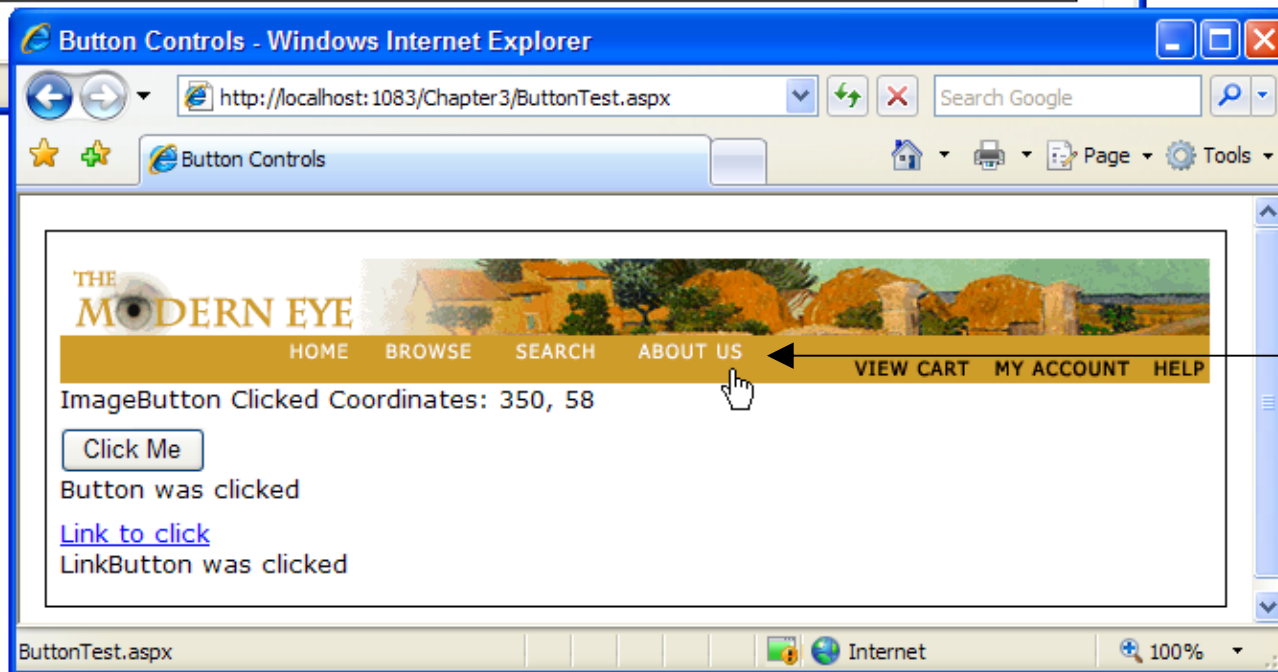
RANDY CONNOLLY

# Button-Style Controls



Button

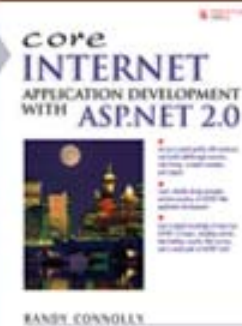
LinkButton



ImageButton

## Button-Style Controls

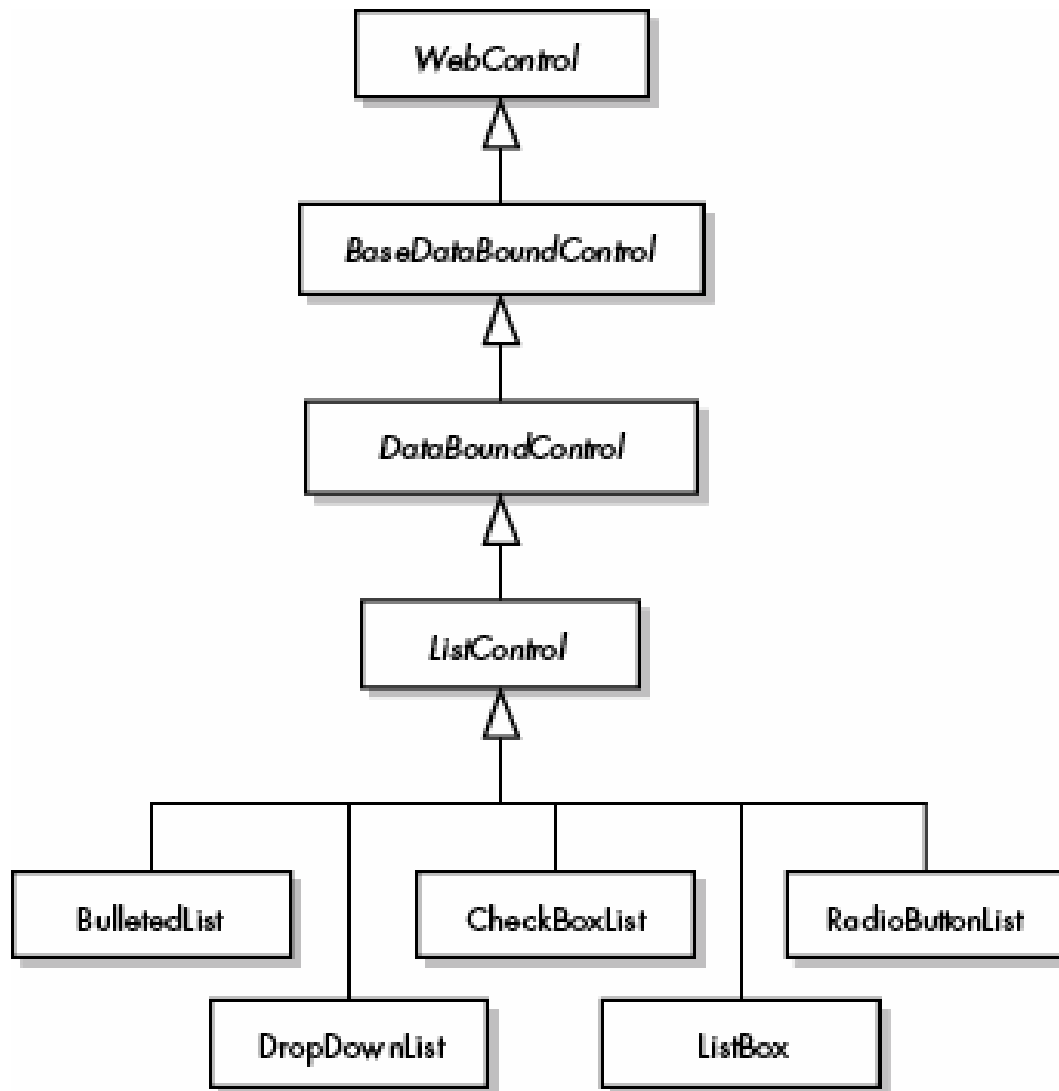
- Support two main events. Both are raised when user clicks on the button.
  - Click
    - Used for single buttons
  - Command
    - Used when you want several related buttons to share the same event handler.



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# List-Style Controls



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

Core  
INTERNET  
APPLICATION DEVELOPMENT  
WITH ASP.NET 2.0



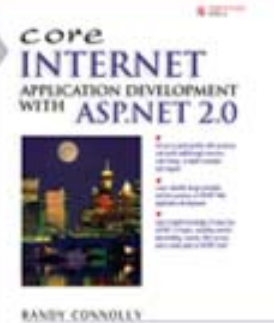
RANDY CONNOLLY

# Key Properties of ListControl

DataTextField	Specifies the field name of the data source that will provide the textual content for the list.
DataTextFormatString	Specifies the formatting string that controls the visual display of the list content.
DataValueField	Specifies the field name of the data source that will provide the value for each list item.
Items	The collection of ListItems in the control.
SelectedIndex	The index (starting with 0) of the selected list item(s).
SelectedItem	The list item that was selected.
SelectedValue	The value of the list item that was selected

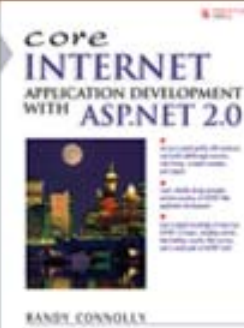
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)



# HyperLink vs LinkButton

- Both are visually displayed in the browser as a hyperlink (i.e., the <a> tag).
- HyperLink
  - simply a link that can be programmatically manipulated.
  - Does not generate a postback.
- LinkButton
  - Generates a postback and thus supports postback event handling (e.g., OnClick).



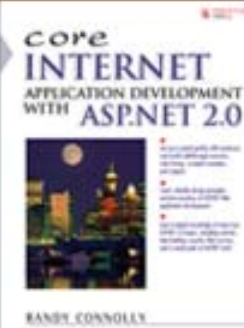
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Table Control

- The `Table` web server control is used for creating server-programmable tables.
- It is usually easier to use the HTML `<table>` element for static tables.
  - The `Table` server control can be used to dynamically add rows or columns to a table at run-time.

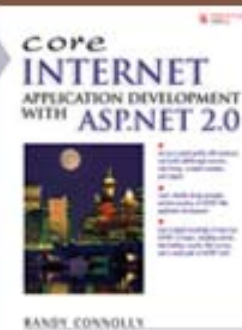


CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Table Control

- The Table control can also make it easier to create accessible tables.
  - ASP.NET 2.0 now supports those features in HTML 4.0 that help those who use screen readers or assistive technologies to understand a web page's tables.



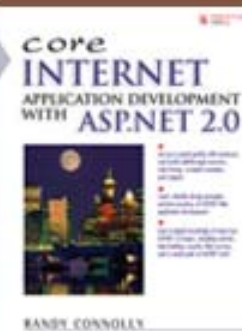
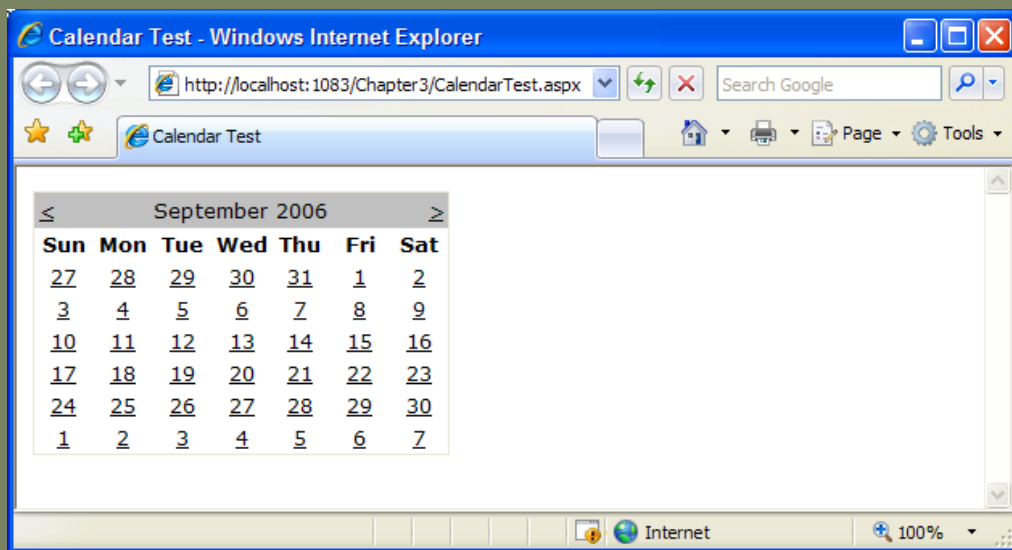
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Calendar Control

- The Calendar control is perhaps the most complex of the basic server controls covered in this chapter.
- It displays a single month calendar that allows a user to navigate from month to month and to select dates, weeks, or entire months.



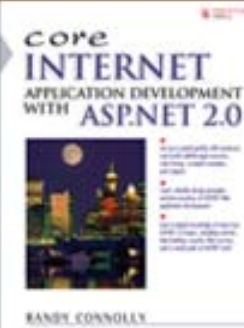
CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007  
[www.randyconnolly.com/core](http://www.randyconnolly.com/core)

# Style Elements

- The appearance of the Calendar control can be further customized by setting various **style elements**.
- Many of the more complex web server controls in ASP.NET use style elements.
  - These are additional tags that are embedded within the parent control and which allow you to customize the appearance of that control.

```
<asp:calendar id="calTest" runat="server">  
  <titlestyle font-size="14px" font-bold="true" />  
</asp:calendar>
```



CORE INTERNET  
APPLICATION DEVELOPMENT  
WITH  
ASP.NET 2.0

Prentice Hall, 2007

[www.randyconnolly.com/core](http://www.randyconnolly.com/core)