

ASP.NET...i

Dr. Binu P Chacko

Associate Professor

Department of Computer Science

Prajyoti Niketan College, Pudukad, THRISSUR

Introduction

- Part of .NET framework...for web applications
- .NET Framework 1.0 released in 2002
- .NET 1.1 in 2002
- IDE Visual Studio .NET in 2003
- ASP.NET 2.0 and Visual Studio 2005
- ASP.NET 3.5 and Visual Studio 2008
- ASP.NET 4 and Visual Studio 2010
- Visual Web Developer is used to create ASP.NET web sites
- 2 *versions*: part of Visual Studio 2010 or Microsoft Visual Web Developer 2010 Express

Creating website

- Start **VWD 2010** from Windows Start menu
- Choose **Web Development** settings from the dialog box
- Choose **New Web Site** from File menu
- Choose Visual Basic or Visual C# from **Installed Templates** section
- Choose **ASP.NET Web Site**
- Choose File System from **Web location**
- Click **OK**

Cont...

- Opens the file `Default.aspx`

```
<asp:Content ID="BodyContent" runat="server"  
  ContentPlaceHolderID="MainContent">
```

```
  <h2>Welcome</h2>
```

```
  <p>to the World of ASP.NET <%=  
  DateTime.Now.ToString() %></p>
```

```
</asp:Content>
```

- Press `Ctrl+F5` to open the browser

IDE

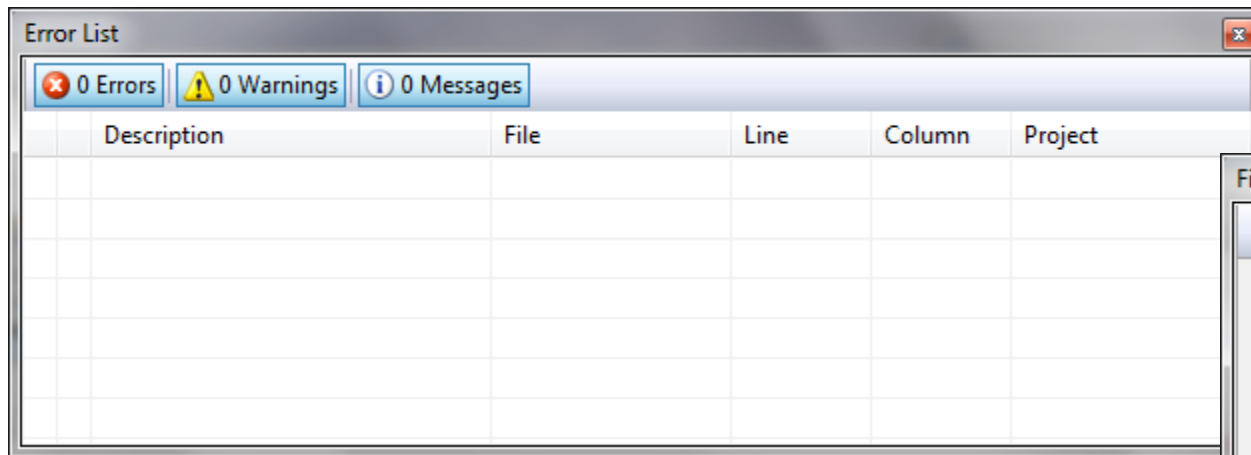
- Main menu
- Toolbar area
- Toolbox
- Document window
- Solution explorer: gives an overview of files
 - Add, move, delete, and rename files
 - View->Solution Explorer
- Database explorer (Server Explorer)
- Properties grid: Press F4

Toolbox

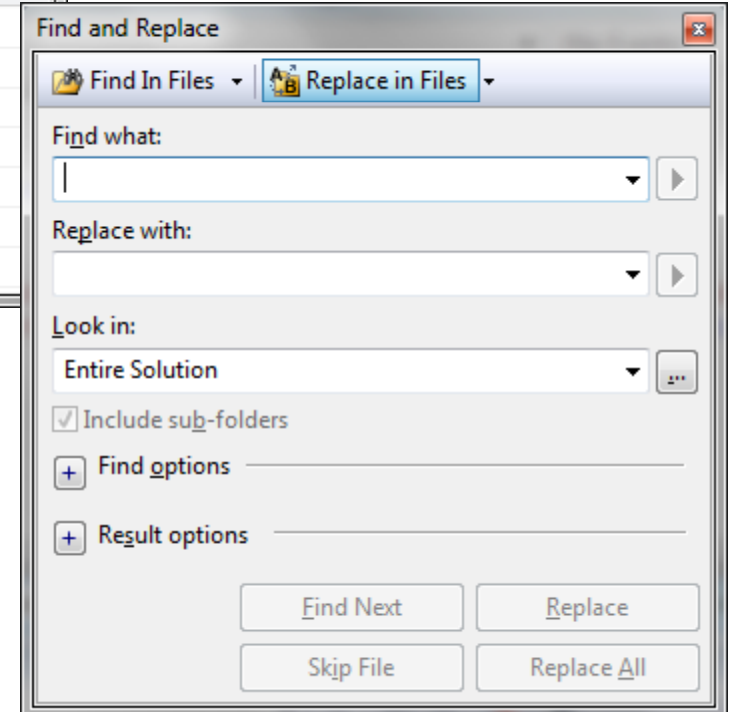
- Create an Empty Web Site
- Change to Design View
- Drag & drop Textbox and Button from Toolbox (Ctrl+Alt+X)
- Press Ctrl+F5 to view in browser

Informational Windows

- Select from **View** menu
- **Error List**: Shows incorrect markup in ASPX and XHTML, errors in VB, C#, XML and CSS files



- **Output Window**
- **Find and Replace**:
Press Ctrl+Shift +H



Customizing IDE

- **Window**: floating, dockable, tabbed document, hide, auto hide
- Drag the window and drop it by clicking the arrow
- **Modifying Toolbox**: Choose Add Tab from Toolbox (right click)
- Drag & drop the item in this tab and Rename Item (right click)
- *Code formatting*: Tools->Options: Text Editor->HTML->Format: Tag Specific Options

Cont...

Document window *changing ...*

- **Fonts & colours:** Tools->Options: Environment
->Fonts and Colors
- **Tab size:** Tools->Options: Text Editor->All
languages->Tab size
- **Line breaks:** Tools->Options: Text Editor->
HTML->Format->Tag Specific Options: Default
settings

Cont...

Toolbars

- **Enable/Disable**: RC on a toolbar-> check/uncheck a toolbar in the list
- **Edit/New**: RC on a Toolbar->Customize [Toolbars tab, Commands tab]

Keyboard shortcut: Tools->Options->Environment:
Keyboard (double shortcut key)

Reset

- **Window**->Reset Window Layout
- **Toolbox**: RC on the Toolbox->Reset Toolbox
- **All**: Tools->Import and Export Settings

Web site project

- File->New->Web Site
- Web site templates: ASP.NET web site, ASP.NET empty web site
- Windows Communication Foundation (WCF) Service: enables to create methods that are callable over a network
- Dynamic Data web sites: to manage data in a database without a lot of manual code
- Opening web site: File->Open->Web Site

File types

- Website->Add New Item

Web files

File type	Extension	Description
Web form	.aspx	Workhorse of ASP.NET web site
Master page	.master	Define global structure
Web user control	.ascx	Contains page fragments that can be used in multiple pages in the web site
HTML page	.htm/.html	To display static HTML
Style sheet	.css	Contains CSS code to style and format the web site
Web configuration file	.config	Contains global configuration information
Site map	.sitemap	Contains hierarchical representation of file in the web site in XML format
JScript file	.js	Contains JavaScript
Skin file	.skin	Contains design information for controls in the web site

Code files & Data files

File type	Extension	Description
Web service	.asmx	Can be called by other systems, and contain code that can be executed on a server
Class	.cs / .vb	Contain code to program the web site
Global application class	.asax	Contian code that is fired on events such as error or the start of an application
XML file	.xml	Used to store data in XML format
SQL server database	.mdf	Databases used by MS SQL server
ADO.NET entity data model	.dbml	Used to access databases declaratively, without the need to write code

Working on SE

RC on website in SE, and then choose ...

- Create Master Page: **Add New Item**
- **Add Existing Item**
 - click on Refresh in SE to get the file list
- **New Folder**

Web forms

- **Views:** Source (default), Design, Split
- Tools->Options: HTML Designer->General
- **Code behind file & Inline file** (uncheck *Place Code in Separate File* while creating the web form)
- Place a Label in the form
- DC on white area in Design view

Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

Label1.Text = "Date & Time : " & DateTime.Now.ToString()

End Sub

- Ctrl+Spacebar to invoke **IntelliSense**
- RC on web site->choose **Start Options**
- RC on .aspx file->choose (**Set as Start Page, View in Browser**)

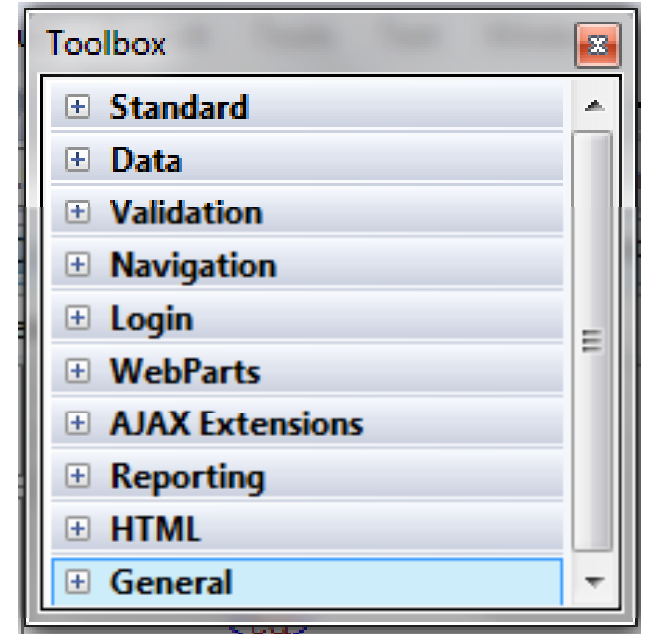
Working in Design view

- **Table**->Insert Table
View->Visual Aids->Visible Borders
- Format->**Bullets and Numbering**
- Connecting pages with **Convert to Hyperlink**

common properties of Controls

Property	Description
AccessKey	Set a key with which a control can be accessed
BackColor ForeColor	To specify background and text colour
BorderColor BorderStyle BorderWidth	Border specifications
CssClass	HTML class points to a CSS class
Enabled	Determines whether the user can interact with the control in the browser
Font	To define font related settings
Height Width	Specify height and width of the control
TabIndex	Determines the order in which users can move through the controls in the page
ToolTip	To set a tooltip for the control
Visible	Determines whether or not the control is sent to the browser

Types of controls



Standard controls

- **Simple controls:** TextBox, Button, Label, HyperLink, RadioButton, CheckBox
- **List controls:** ListBox, DropDownList, CheckBoxList, RadioButtonList, BulletedList

Textbox, checkboxlist, dropdownlist, label

- C & D `DropDownList` into the Design view
- C & D `CheckBoxList` into the Markup view
- C & D `Button` and `Label` into the Design view
- DC on the `Button` and enter the following code into the code behind file

```
Label1.Text = "You selected DDL " & DropDownList1.SelectedValue  
& "<br/>"
```

```
For Each item As ListItem In CheckBoxList1.Items
```

```
    If item.Selected Then
```

```
        Label1.Text &= "You selected CBL " & item.Value & "<br/>"
```

```
    End If
```

```
Next
```

- Click `Save All` and `View in Browser`

Container controls

- Panel, PlaceHolder, MultiView, Wizard
- To group related content and controls
- Place **CheckBox** and **Panel** controls on the Design view
- Text : Show Panel, AutoPostBack: True
- `Panel1.Visible = CheckBox1.Checked`
- **Visible : False**
- *Type Visibility checking*
- Press Ctrl+F5

Wizard control

- To break large page into multiple areas
- Place **Wizard** inside the panel. Drag its right edge to 500px width
- Select **Add/Remove WizardSteps** from its smart task panel
- Create steps with **Titles** About you, Favourite language, Ready
- Select **StepTypes** Auto, Finish and Complete
- `ActiveStepIndex : 0`
- Click on About you, and place a **Label** and **TextBox** on the grey rectangle
- Label – Text : Name, TextBox – ID: YourName
- Click on Favourite language and place a **DropDownList** in the rectangle
- Click on Ready, and place a **Label** (ID: Result) in the rectangle
`Result.Text = "Name : " & YourName.Text`
`Result.Text &= "
Favourite Language : " & FavaouriteLanguage.SelectedValue`
- Press Ctrl+F5

State Engine

- Insert a **Table** of size 2 x 2
- Place **Label** and **Calendar** (choose *Simple* from *Auto Format*) in the 1st column
- Place **Button1** (ID: SetDate, Text: Date) and **Button2** (Text: Post Back) in the 2nd column
Label1.Text = DateTime.Now.ToString
- Press Ctrl+F5
- EnableViewState: False (**Label**)
- Press Ctrl+F5

Data types

.NET	VB.NET	Description
System.Byte	Byte	0 to 255
System.Int16	Short	-32,768 to 32,767 0
System.Int32	Integer	-2,147,483,648 to 2,147,483,647
System.Int64	Long	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
System.Single	Single	-3.4028235E+38 to 3.4028235E+38
System.Double	Double	
System.Decimal	Decimal	
System.Boolean	Boolean	True or False
System.DateTime	Date	1/1/0001: 12.00 am
System.Char	Char	Nothing
System.String	String	Up to 2M characters (Nothing)
System.SByte	SByte	-128 to 127
System.UInt16	UShort	0 to 65,535
System.UInt32	UInteger	0 to 4,294,967,295
System.UInt64	Ulong	0 to 18,446,744,073,709,551,615
System.Object	Object	Parent of all data type (Nothing)

Converting & casting data types

- Dim myboolean1 As boolean = Convert.ToBoolean("True")
- ToInt32, ToDateTime, ToString
- Dim o1 As Object = 1
- Dim i1 As Integer = DirectCast(o1, Integer)
- Dim i2 As Integer = CType(o1, Integer)
- Dim o2 As Double = 1
- Dim i3 As Integer = DirectCast(o2, Integer) fails
- Dim i4 As Integer = CType(o2, Integer)
- CType allows to cast between two objects that look similar
- DirectCast allows to cast between compatible types
- TryCast returns Nothing if casting is not possible, whereas DirectCast and CType will crash the code

Arrays

- Dim roles(1) As String upper bound
- roles(0) = "Administrator"
- roles(1) = "ContentManager"
- ReDim Preserve roles(2) redimension the array
- Roles(2) = "Member"
- Preserve keep the previous values

Collections

- Dim roles As New ArrayList()
- roles.Add("Administrator")
- ArrayList is not strongly typed. It contains multiple types of objects
- Solution: Generics
- Generics allow to write code template that can be used in different scenarios with different types
- Dim roles As New List(Of String)
- Roles.Add("Administrator")
- Collection initializer
- Dim myList As New List(Of Integer) From {1, 2, 3, 4, 5}

Operators

- Assignment operator
- Dim age as Integer = 35
- Arithmetic operators: +, -, *, /, \ (integer division), ^, Mod
- Relational operators: =, <>, <, >, <=, >=, Is (compares two objects)
- Concatenation operators: &, &=
- Option Strict On to stop implicit conversion
- Logical operators: And, Or, Not, AndAlso, OrElse

Decision making statements

```
If n1 = 3 AndAlso n2 = 4 Then
    .....
End If
```

```
If <condition> Then
    ....
Else
    ....
End If
```

```
Select Case <variable>
    Case <value 1>
        .....
    Case <value 2>
        .....
    Case Else
        .....
End Select
```

```
If <condition> Then
    ...
Elseif <condition> Then
    ....
Else
    .....
End If
```

Loops

' Syntax of
"" loop statements

```
For loopCount As Integer = 0 To 10 Step 2  
    .....  
Next
```

```
For Each role As String In roles  
    .....  
Next
```

```
While <condition>  
    .....  
End While
```

Calculator

- Insert a 3 x 3 **Table** in the Design view
- Merge the 1st row: RC on selected row, choose Modify-> Merge Cells
- Place a **Label** (ID: ResultLabel, Text: Clear the test by RC on it and choose Reset)
- **TextBox1** (ID: ValueBox1), **TextBox2** (ID: ValueBox2)
- **DropDownList** (ID: OperatorList), **Button** (ID: CalculateButton, Text: Calculate)
- Press Ctrl+F5

```
If ValueBox1.Text.Length > 0 AndAlso ValueBox2.Text.Length > 0 Then
    Dim result As Double
    Dim value1 As Double = Convert.ToDouble(ValueBox1.Text)
    Dim value2 As Double = Convert.ToDouble(ValueBox2.Text)
    Select Case OperatorList.SelectedValue
        Case "+"
            result = value1 + value2
        Case "-"
            result = value1 - value2
        Case "*"
            result = value1 * value2
        Case "/"
            result = value1 / value2
    End Select
    ResultLabel.Text = result.ToString
Else
    ResultLabel.Text = String.Empty
End If
```

Calculator class

- RC on web site folder, choose ASP.NET Folder->App_Code
- RC on App_Code folder and choose Add New Item
- Select Class with name Calculator

```
Public Function Add(ByRef a As Double, ByRef b As Double) As Double
    Return a + b
End Function
Public Function Subtract(ByRef a As Double, ByRef b As Double) As Double
    Return a - b
End Function
Public Function Multiply(ByRef a As Double, ByRef b As Double) As Double
    Return a * b
End Function
Public Function Divide(ByRef a As Double, ByRef b As Double) As Double
    Return a / b
End Function
```

Cont...

- Button in CalculatorDemo.aspx

```
If ValueBox1.Text.Length > 0 AndAlso ValueBox2.Text.Length > 0 Then
    Dim result As Double
    Dim value1 As Double = Convert.ToDouble(ValueBox1.Text)
    Dim value2 As Double = Convert.ToDouble(ValueBox2.Text)
    Dim myCalculator As New Calculator
    Select Case OperatorList.SelectedValue
        Case "+"
            result = myCalculator.Add(value1, value2)
        Case "-"
            result = myCalculator.Subtract(value1, value2)
        Case "*"
            result = myCalculator.Multiply(value1, value2)
        Case "/"
            result = myCalculator.Divide(value1, value2)
    End Select
    ResultLabel.Text = result.ToString
Else
    ResultLabel.Text = String.Empty
End If
```

OOP

Object

- Dim myCalculator As Calculator = New Calculator()
- Dim age As Integer = New Integer()

Class

```
Public Class ClassName
```

```
.....
```

```
End Class
```

Property: Characteristics of the object

```
Public Class Person
```

```
Private _firstName As String
```

```
Public Property FirstName() As String
```

```
Get
```

```
Return _firstName
```

```
End Get
```

```
Set (ByVal value As String)
```

```
_firstName = value
```

```
End Set
```

```
End Property
```

```
End Class
```

Backing variable

```
Dim myPerson As Person = New Person()
```

```
myPerson.FirstName = "Deon"
```

```
' Accessing setter
```

```
Label1.Text = myPerson.FirstName
```

```
' Accessing getter
```


Cont...

Read only/Write only Property

```
Public ReadOnly Property FullName() As String
```

```
    Get
```

```
        Return _firstName & " " & _lastName
```

```
    End Get
```

```
End Property
```

- Method

```
Public Class Person
```

```
    Public Sub Save()
```

```
        .....
```

```
    End Sub
```

```
End Class
```

```
Dim myPerson As Person = New Person()  
myPerson.FirstName = "Leo"  
myPerson.Save()
```

Cont...

- **Constructor**

```
Public Class Person
```

```
    Public Sub New(ByVal firstName As String, ByVal lastName As String, ByVal  
        dateOfBirth As DateTime)
```

```
        _firstName = firstName
```

```
        _lastName = lasName
```

```
        _dateOfBirth = dateOfBirth
```

```
    End Sub
```

```
End Class
```

```
Dim myPerson As Person = New Person("Ijas", "Ahamed", New  
    DateTime(1980, 7, 2))
```

OR

```
Dim myPerson As New Person("Ijas", "Ahamed", New DateTime(1980, 7, 2))
```

Object Initiatizers

```
Dim myPerson As New Person() With {.FirstName = "Ijas", .LastName =  
    "Ahamed"}
```

Cont...

- Inheritance

Public Class **Student**

Inherits **Person**

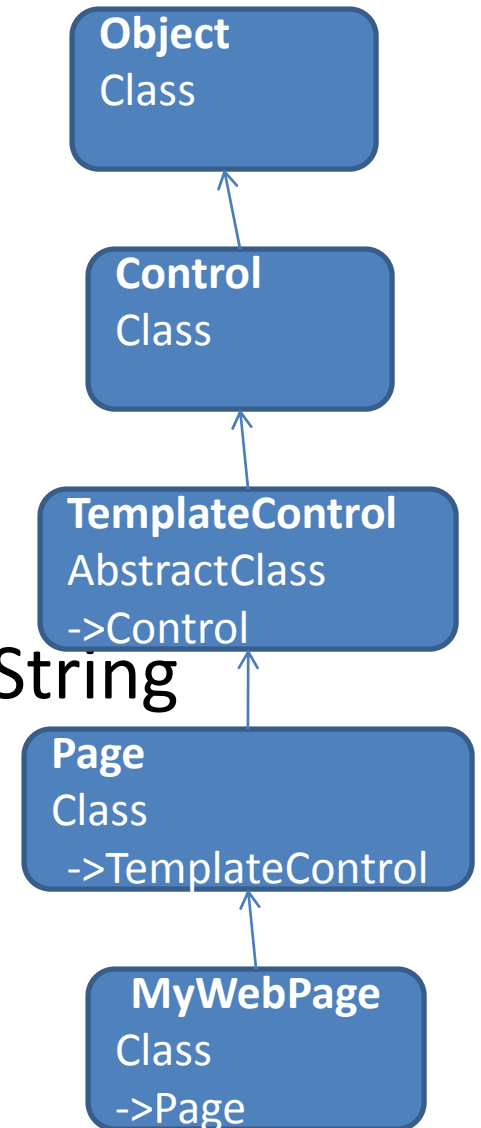
Function overriding

Public **Overrides** Function ToString() As String

Return FullName & ", born at " &
_dateOfBirth.ToShortDateString()

End Function

Label1.Text = myPerson.ToString()



Events

- Raised by certain objects in the application and then handled by others
- Write event handler

```
Protected Sub Button1_Click(ByVal sender As Object,  
    ByVal e As System.EventArgs) Handles Button1.Click
```

```
End Sub
```

event wiring code

- Pass information to event handler using event argument class (System.EventArgs)
- sender contains a reference to the object that triggered the event (Button1)

Master page

- To create web pages with a consistent layout
- Cannot be request in a browser directly
- Template for content pages
- At runtime, the markup from master and content pages are merged, processed and send to the browser
- Content pages are connected master page using

```
<%@ Page Language="VB"  
    MasterPageFile="~/MasterPages/Frontend.master"  
    AutoEventWireup="false" CodeFile="Welcome.aspx.vb"  
    Inherits="Welcome" title="Untitled Page" %>
```

- Page specific content is put inside a **Content** control

```
<asp:Content ID="Content1" ContentPlaceHolderID="head"  
    Runat="Server">
```

```
</asp:Content>
```

Cont...

```
<%@ Master Language="VB"  
  CodeFile="Frontend.master.vb"  
  Inherits="MasterPages_Frontend" %>
```

- To create regions that content pages can fill in

```
<asp:ContentPlaceHolder  
  id="ContentPlaceHolder1" runat="server">
```

```
</asp:ContentPlaceHolder>
```

Cont...

- Create the folder [Master Pages](#) in the Website
- Create the master page [Frontend.master](#) by RC on the folder->Add New Item->Master Page
- Modify `<form>` in master page

```
<form id="form1" runat="server">
  <div id = "PageWrapper">
    <div id = "Header"><a href="~/ " runat="server">Header section</a></div>
    <div id="MenuWrapper">Main Menu</div>
    <div id="MainContent">
      <asp:ContentPlaceHolder id="ContentPlaceHolder1" runat="server">

        </asp:ContentPlaceHolder>
      </div>
    <div id="Sidebar">Sidebar section</div>
    <div id="Footer">Footer section</div>
  </div>
</form>
```

Cont...

- Check **Select master page** while creating the content page
- Choose the master page from **Select a Master Page** dialog box
- Press Ctrl+F5
- Create another content page
- Go back to the **Design** view of 1st content page
- Create a link to 2nd content page
- Press Ctrl+F5

Base page

- Create a Class file BasePage in App_Code folder

Public Class BasePage

Inherits System.Web.UI.Page

Private Sub Page_PreRender(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.PreRender

 If Me.Title = "Untitled Page" Or String.IsNullOrEmpty(Me.Title)
 Then

 Throw New Exception("Page title cannot be ""Untitled Page"" or
 an empty string")

 End If

End Sub

End Class

- Modify to Inherits BasePage in Login.aspx.vb
- Press Ctrl+F5. *Error*
- Give a Title in @ Page directive in Login.aspx
- Press Ctrl+F5

Page life cycle

Phase	Description
Page request	A request to an ASPX page starts the life cycle of that page
Start	Page gets access to properties like Request and Response . PreInit event is raised
Page initialization	Controls set up in the page or added programmatically become available. Page class fires the events Init , InitComplete , PreLoad . Control properties are loaded from View State and Control State again during a postback
Load	Page raises Load event
Validation	Validation controls are processed
Postback event handling	Controls in the page may raise their own events
Rendering	Controls and the page itself output their HTML to the browser
Unload	Clean-up phase. Page and controls can release resources like database connections. Unload event is raised

Page Template

- Add a **Web Form** (Temporary.aspx) to the site based on a master page
- Modify in the Markup view

Inherits = “\$relurlnamespace\$_\$safeitemname\$”

- Modify Code Behind file

Partial Class \$relurlnamespace\$_\$safeitemname\$

Inherits BasePage

- **Save All** and choose **File->Export Template**
- Choose **Item Template** and programming language. Click Next
- Check Temporary.aspx. Click Next
- Click Next to leave **Select Item References**
- Give the Template name (MyBasePage) and description in **Select Template Options**. Click Finish
- A ZIP file creates
- **Add New Item** (TestPage.aspx) to the site based on the template MyBasePage
- \$relurlnamespace\$_\$safeitemname\$ is renamed to **_TestPage** in Markup view and Code Behind file

Themes

- A collection of files that define the look of a page
- Includes skin files, CSS files and images
- Define themes in `App_Themes` folder
RC on Website, Choose Add ASP.NET Folder->Theme
- For each theme, create a sub folder with theme's name
- Types of themes: `Theme`, `StyleSheetTheme`

Applying themes

at page level

```
<%@ Page Language="VB" AutoEventWireup="false"  
    CodeFile="Default.aspx.vb" Inherits="_Default" Theme="DarkGrey" %>
```

- Replace *Theme* with *StyleSheetTheme* to select the other property
- Modify web.config

at the site level

```
<pages theme="DarkGrey" [stylesheetTheme="DarkGrey"]> .....  
</pages>
```

- Set themes programmatically

Dynamically Switching Themes

- Open the master page in Markup view and place a [DropDownList](#) between <div> tags

ID: ThemeList

- Open [Smart Task](#) panel in Design view

[Enable AutoPostBack](#)

Insert items Monochrome and DarkGrey using [Edit Items](#)

- DC the [DropDownList](#) and add the following code

```
Protected Sub ThemeList_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles ThemeList.SelectedIndexChanged
```

```
    Dim preferredTheme As HttpCookie = New HttpCookie("PreferredTheme")
```

```
    preferredTheme.Expires = DateTime.Now.AddMonths(3)
```

```
    preferredTheme.Value = ThemeList.SelectedValue
```

```
    Response.Cookies.Add(preferredTheme)
```

```
    Response.Redirect(Request.Url.ToString())
```

```
End Sub
```

- The code retrieves the selected theme from the list and stores it in a cookie

Cont...

- DC the page and add the following code

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
```

```
    If Not Page.IsPostBack Then
```

```
        Dim selectedTheme As String = Page.Theme
```

```
        Dim preferredTheme As HttpCookie = Request.Cookies.Get("PreferredTheme")
```

```
        If preferredTheme IsNot Nothing Then
```

```
            selectedTheme = preferredTheme.Value
```

```
        End If
```

```
        If Not String.IsNullOrEmpty(selectedTheme) AndAlso ThemeList.Items.FindByValue(selectedTheme) IsNot Nothing Then
```

```
            ThemeList.Items.FindByValue(selectedTheme).Selected = True
```

```
        End If
```

```
    End If
```

```
End Sub
```

- Click [Save All](#) and browse Default.aspx
- The item you chose last in the drop down list will be preselected in the drop down list

PreInit event

- Open the BasePage and add the following code that sets the selected theme during **PreInit** event

```
Private Sub Page_PreInit(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.PreInit
```

```
    Dim preferredTheme As HttpCookie = Request.Cookies.Get("PreferredTheme")
```

```
    If preferredTheme IsNot Nothing Then
```

```
        Page.Theme = preferredTheme.Value
```

```
    End If
```

```
End Sub
```

- Click **Save All** and browse Default.aspx

User Controls

- Create folders Controls, Images to the Website. Place images into Images folder
- RC Controls folder and choose [Add New Item](#). Select [Web User Control](#) and give its name as Banner(.ascx)
- Place a [Panel](#) in the Design view. ID: VerticalPanel
- Place [Image](#) control inside the Panel. Use [ImageUrl](#) property to select an image from Images folder.
[AlternateText](#): Sample banner
- Place the *image* tag within *a* tags.
href=<http://p2p/wrox/com> target="_blank"
- Save the file. Close it (Ctrl+F4)

Adding user control to a page

- Place Banner.ascx into the [master page](#) (Frontend.master) in Design view
- Modify [@ Register](#) directive in Markup view

```
<%@ Register src="~/Controls/Banner.ascx"  
  TagName="Banner" TagPrefix="uc1" %>
```
- Save and close the master page
- Browse [Default.aspx](#). Banner displays along with other contents in the page
- Click on the image to view it in a separate window

Site wide registration

- Open [web.config](#) file
- Locate `<pages>` element, and add the following code

```
<pages theme="Monochrome">
```

```
  <controls>
```

```
    <add targePrefix="Wrox" tagName="Banner"  
    src="~/Controls/Banner.ascx" />
```

```
  </controls>
```

```
</pages>
```

- Save and close the file
- Open [master page](#) in Markup view and locate `Banner` control.
Change `uc1` to `Wrox`

```
<Wrox:Banner ID="Banner1" runat="server" />
```

- Remove `@ Register` directive
- Save and close the master page