

1. Image Control:

- **Purpose:** Image control is a server side control. So we can use this control to load the dynamic image control. We can also load the server side image or static image in this control.
- **Properties:**

AutoPostBack	To raise the event automatically, set property true.
DataTextField	At the time of server connectivity, if data loads into this control from the database then set data text field property, this data will be displayed to the user.
DataValueField	At the time of server connectivity, if data loads into this control from the database then set data value field property, this data will be hide from the user.
Item	To add data, this property is used. To display static items, use this property.
-List Item:	List item can be added from the item property.
Text	Each list item property has text field, the text field will be displayed to the user.
Value	Each list item property has text field, the value field will be hide from the user.

- **Events:** Selected Index Changed

2. File Upload:

- **Purpose:** to browse the file from the client pc.
- **Properties:** generally no need to set its properties.
- **Events:** No Event

3. Drop Down List:

- **Purpose:** To store multiple items and to select single item. Default it display single item. If you want to see more items, click on the drop down.

- Properties:

AutoPostBack	To raise the event automatically, set property true.
DataTextField	At the time of server connectivity, if data loads into this control from the database then set data text field property, this data will be displayed to the user.
DataValueField	At the time of server connectivity, if data loads into this control from the database then set data value field property, this data will be hide from the user.
Item	To add data, this property is used. To display static items, use this property.
-List Item:	List item can be added from the item property.
Text	Each list item property has text field, the text field will be displayed to the user.
Value	Each list item property has text field, the value field will be hide from the user.

- Events:Selected Index Changed

4. Bulleted List:

- Purpose:To display the item like list, use the bulleted list. Bulleted list control has the facility to display the items with bullets.
- Properties:

AutoPostBack	To raise the event automatically, set property true.
BulletImageUrl	To load the image in the place of bullet, set the path of the image.
BulletStyle	We can set the different bullet style, like Numbered, Lower Alpha, Upper Alpha, etc.
DisplayMode	We can set the data's display mode like Text, Hyperlink or Button.

TextField	At the time of server connectivity, if data loads into this control from the database then set data text field property, this data will be displayed to the user.
DataValueField	At the time of server connectivity, if data loads into this control from the database then set data value field property, this data will be hide from the user.
FirstBulletNumber	If we have select the Bullet Style like Numbered, Lower Alpha, Upper Alpha then we can set this property.
Item	To add data, this property is used. To display static items, use this property.
-List Item:	List item can be added from the item property.
Text	Each list item property has text field, the text field will be displayed to the user.
Value	Each list item property has text field, the value field will be hide from the user.

- **Events:** Bulleted List Click event

5. Check Box:

- **Purpose:**select multiple checkbox items, we can use this control.
- **Properties:**

AutoPostBack	To raise the event automatically, set property true.
Checked	If we want to set default checked checkbox then set its property true.
TextAlign	We can set the text alignment using this property.

- **Events:** CheckedChanged

6. Radio Buttons:

- **Purpose:**to select 1 item from the multiple items.

- Properties:

AutoPostBack	To raise the event automatically, set property true.
Checked	If we want to set default checked checkbox then set its property true.
GroupName	Set the same group name of particular group of radio buttons like gender group.
TextAlign	We can set the text alignment using this property.

- Events: CheckedChanged

7. Hyperlink:

- Purpose: to navigate from one page to another page.
- Properties:

ImageUrl	To display link on image, then set the path of the image.
NavigateUrl	Give the path of the page, to which you want to navigate.
Text	Set the Link text, on which user click and user navigate to other page.
Target	Where the page should open, like in other tab, other window or on the same page.

8. Tables:

- Purpose: to display dynamic or static rows and columns.
- Properties:

BackImageUrl	To set background image in the table.
Caption	To set the name of the table.
CaptionAlign	To set the alignment of the caption like top, bottom, left or right.
GridLines	To show/hide gridlines from the table.

HorizontalAlign	To set the horizontal alignment of the table.
Rows	To add the rows and cells statically in the table.

9. Panel:

- **Purpose:** it is a container control. To display the content at specific part of screen, it will be easy using panel. To display or hide some part, it is easy using panel. To display the big content in the specific area of the screen it is helpful using scrollbars.
- **Properties:**

BackImageUrl	To set background image in the table.
Direction	To set the direction inside the panel like right to left, left to right, etc.
GroupingText	To set the caption of the panel.
Height	To set the height of the panel.
HorizontalAlign	To set the horizontal alignment of the panel like left, center, right or justify
Scrollbars	To display the scrollbars like horizontal, vertical, both or Auto.
Width	To set the width of the panel.

10. Tree View control:

The TreeView Web server control is used to display hierarchical data, such as a table of contents or file directory, in a tree structure.

TreeView Node Types

The TreeView control is made up of one or more nodes. Each entry in the tree is called a node and is represented by a TreeNode object. The following table describes the three different node types.

Node type	Description
Root	A node that has no parent node and one or more child nodes.
Parent	A node that has a parent node and one or more child nodes.
Leaf	A node that has no child nodes.

Properties

Name	Description
LeafNodeStyle	Gets a reference to the TreeNodeStyle object that allows you to set the appearance of leaf nodes.
Nodes	Gets a collection of TreeNode objects that represents the root nodes in the TreeViewcontrol.
NodeStyle	Gets a reference to the TreeNodeStyle object that allows you to set the default appearance of the nodes in the TreeView control.
ParentNodeStyle	Gets a reference to the TreeNodeStyle object that allows you to set the appearance of parent nodes in the TreeView control.
PathSeparator	Gets or sets the character that is used to delimit the node values that are specified by theValuePath property.
RootNodeStyle	Gets a reference to the TreeNodeStyle object that allows you to set the appearance of the root node in the TreeView control.
SelectedNodeStyle	Gets the TreeNodeStyle object that controls the appearance of the selected node in theTreeView control.
ShowCheckboxes	Gets or sets a value indicating which node types will display a check box in the TreeViewcontrol.

ShowExpandCollapse	Gets or sets a value indicating whether expansion node indicators are displayed.
ShowLines	Gets or sets a value indicating whether lines connecting child nodes to parent nodes are displayed.

Events

Name	Description
SelectedNodeChanged	Occurs when a node is selected in the TreeView control.
TreeNodeCheckChanged	Occurs when a check box in the TreeView control changes state between posts to the server.
TreeNodeCollapsed	Occurs when a node is collapsed in the TreeView control.
TreeNodeExpanded	Occurs when a node is expanded in the TreeView control.

11. Menu Control:

The ASP.NET Menu control allows you to develop both statically and dynamically displayed menus for your ASP.NET Web pages.

Properties

Name	Description
DynamicHoverStyle	Gets a reference to the Style object that allows you to set the appearance of a dynamic menu item when the mouse pointer is positioned over it.
DynamicMenuItemStyle	Gets a reference to the MenuItemStyle object that allows you to set the appearance of the menu items within a dynamic menu.

DynamicMenuItemStyle	Gets a reference to the MenuItemStyle object that allows you to set the appearance of a dynamic menu.
Items	Gets a MenuItemCollection object that contains all menu items in the Menu control.
Orientation	Gets or sets the direction in which to render the Menu control.
PathSeparator	Gets or sets the character used to delimit the path of a menu item in a Menu control.
StaticHoverStyle	Gets a reference to the Style object that allows you to set the appearance of a static menu item when the mouse pointer is positioned over it.
StaticMenuItemStyle	Gets a reference to the MenuItemStyle object that allows you to set the appearance of the menu items in a static menu.
StaticMenuStyle	Gets a reference to the MenuItemStyle object that allows you to set the appearance of a static menu.
StaticSelectedStyle	Gets a reference to the MenuItemStyle object that allows you to set the appearance of the menu item selected by the user in a static menu.

12. Site Map Path Control:

Properties

Name	Description
CurrentNodeStyle	Gets the style used for the display text for the current node.
NodeStyle	Gets the style used for the display text for all nodes in the site navigation path.

PathSeparator	Gets or sets the string that delimits SiteMapPath nodes in the rendered navigation path.
RootNodeStyle	Gets the style for the root node display text.
Style	Gets a collection of text attributes that will be rendered as a style attribute on the outer tag of the Web server control.

XML File Code:

```

<siteMap>
    <siteMapNode url="Home.aspx" title="Root" description="" />
    <siteMapNode url="AboutUs.aspx" title="Parent" description="" />
    <siteMapNode url="ContactUs.aspx" title="Parent" description="" />
    <siteMapNode url="Services.aspx" title="Parent" description="" />
        <siteMapNode url="Electronics.aspx" title="Child" description="" />
    </siteMapNode>
</siteMapNode>
</siteMap>

```

13. Wizard Control:

The main purpose of this control is to execute the process step by step. Using this controls we can take inputs from the user step by step.

Important Properties:

ActiveStepIndex	FinishPreviousButtonStyle
CancelButtonStyle	HeaderStyle
CancelButtonText	HeaderText
CancelButtonType	NavigationButtonStyle
DisplayCancelButton	SideBarButtonStyle

DisplaySideBar	StartNextButtonStyle
FinishCompleteButtonStyle	WizardSteps

Events of Wizard Steps:

ActiveStepChanged	NextButtonClick
CancelButtonClick	PreviousButtonClick
FinishButtonClick	SideBarButtonClick

1. Required Field Validators:

- **Purpose:** This validation control is used to check whether the control is empty or not. Leading and trailing spaces of the input value are removed before validation.
- **Properties:**

ControlToValidate	The id of the control to validate
ErrorMessage	The text to display in the validation summary control and if the text property is not set then this message will be displayed in the validation control
Text	The message to display when validation fails

2. Range Validators

- **Purpose:** This validation control checks that user enter the value falls between two values.

- **Properties:**

ControlToValidate	The id of the control to validate
ErrorMessage	The text to display in the validation summary control and if the text property is not set then this message will be displayed in the validation control
Maximum Value	Specifies the maximum value of the control
Minimum Value	Specifies the minimum value of the control
Type	Specifies what to check. The types are: Currency, date, double, integer, string
Text	The message to display when validation fails

3. Regular Expression validators

- **Purpose:** This control checks that value of an input control matches with specified pattern or not.
- **Properties:**

ControlToValidate	The id of the control to validate
ErrorMessage	The text to display in the validation summary control and if the text property is not set then this message will be displayed in the validation control
Text	The message to display when validation fails
Validation Expression	Specifies the expression used to validate input control.

4. Custom Validations

- **Purpose:** This control allows you to handle the validation using java script or vb script.
- **Properties:**

ControlToValidate	The id of the control to validate
ErrorMessage	The text to display in the validation summary control and if the text property is not set then this message will be displayed in the validation control
Text	The message to display when validation fails
Client Validation Function	Specifies the function name of java script or vb script.

5. Validation Summary

- **Purpose:** It displays the report of all validation controls message in a web page.

- **Properties:**

Display Mode	Specifies How to display summary like BulletedList, List, SingleParagraph
ShowMessageBox	Specifies whether the summary should be displayed in message box or not.

6. Ad Rotators

- **Purpose:** This control is used to display the sequence of the images. This control is using the xml file to store the ad information. The xml file must begin with an <Advertisement> tag. Inside this tag, there may be many <ad> tags which defines each advertisement. The predefined elements in the <ad> tag are listed below:
- **Properties:**

<ImageUrl>	Specifies the path of the image.
<NavigateUrl>	Specifies the navigation path which is optional.
<AlternateText>	Specifies the text when image cannot display.
<Keyword>	Specifies the keyword by which developer can sort the specific type of advertisement.
<Impressions>	Specifies the display rates in percent of hints.

The important property of this control is **AdvertisementFile** to which we have to specify the path of the xml file.

7. Login Controls

These controls provide the facility of the different types of login controls. For these controls, ASP.NET provides the ready coding and designing. We cannot change the coding but we can change the design as per our requirement.

- **Login:**

This control provides the facility to login our web application. We can change the user interface of the control. We can customize the login control as per the user requirement.

- **Create User Wizard**

This control provides the facility to create the user for our web application. Data will be stored automatically in the database. We can customize the login control as per the user requirement.

- **Login View**

This control provides the facility to show the particular message to the particular role person. Like show message to the manager, employee, trainee, etc. we can give specific message to the specific role person. If the manager logs in, in the website then he will get specific message. If anonymous person logs in, he will get the other message. So each role person can different message using the same control. We can set the style to the message.

- **LoginName**

This control provides the facility to show the logged in person name automatically. We can set the specific style to this control.

- **Change Password**

This control provided the facility to change the password. For that user must have to be logged in. We can change the user interface using different properties. We can format the text part as well as control part of this control.

- **Password Recovery**

This control provides the facility to recover the password. In case user forgot his/her password then this control sends the mail to the particular user. We don't need to do code for it. We can format the user interface as per our requirement. To use this control, we must have internet connection with the computer. We must have to do SMTP setting in ASP.NET configuration site.

8. Master Pages and Themes

- **What is master page?**

Using master page, we can create the same look for the all web pages. Some part of the web page may be common for all web pages that should be place in the master page. Master page is working like the template. The main benefit of the master page is that we don't need to design the same thing for all web pages. For example, menu control, header and footer which we require in all web pages. So these controls, we should put in the master page. So that we can give the consistent designing in all web pages.

- **How to create master page?**

To create the master page we should go in add new item->master page and click on add button. Using these steps, we can add the master page.

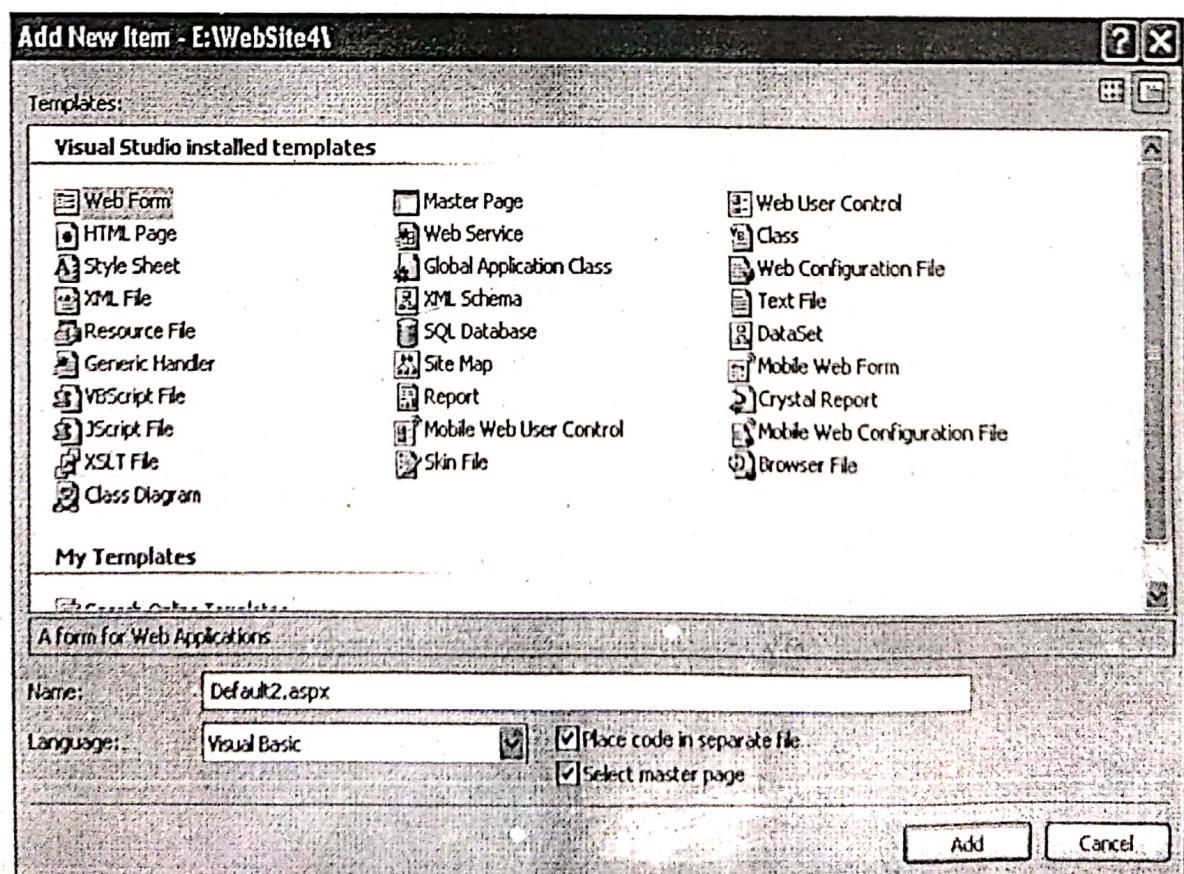
- **What is content place holder?**

Content Place Holder is the place where non common part of our website can be put in this control. This control can be set at Master page. This content place holder, we can use from the each child page. Only content place holder, we can access from the child pages. The part which is out of the content place holder, we can access only from master page. So set this control in the master page and don't put any controls in this control in master page. Use this control only from the child pages.

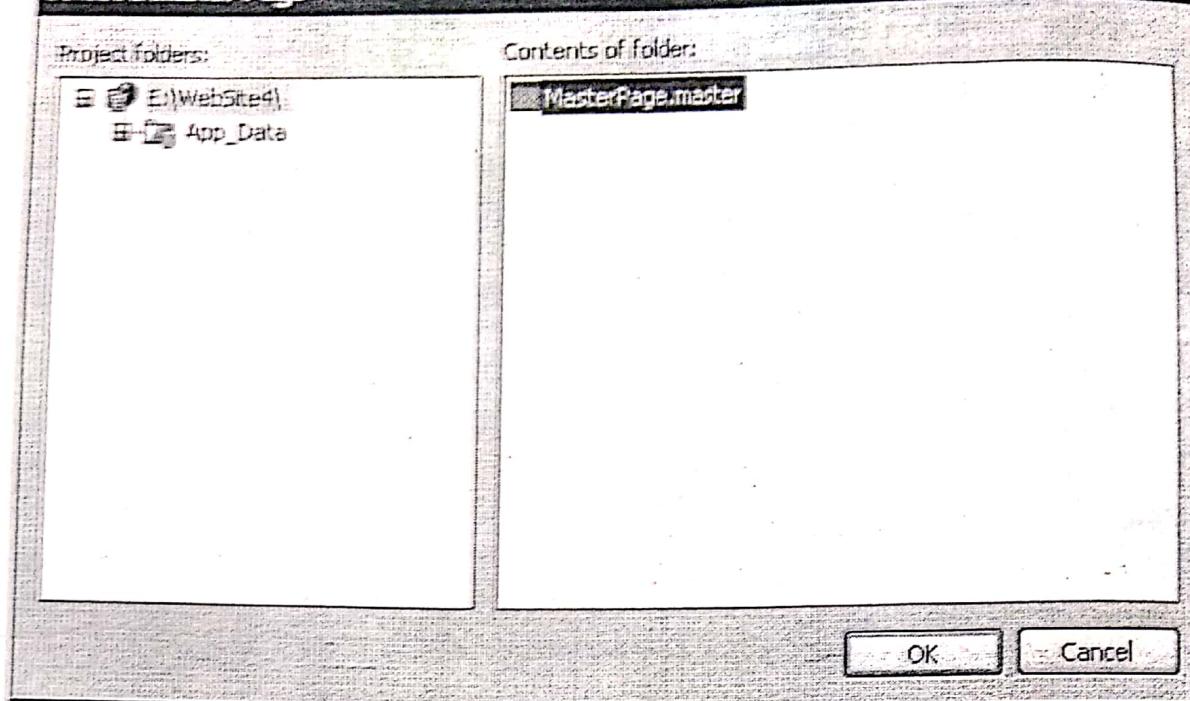
- **How to implement master page to the other page?**

Steps to create child page:

1. Click on add new item, select the web form and click on the check box with the name select master page and click on add.
2. Now select the master page from right side and click on ok button.



Select a Master Page



1. Themes:

• What is Theme?

A theme is a collection of property settings that allow you to define the look of pages and controls, and then apply the look consistently across pages in a Web application, across an entire Web application. For example, we can set the same look of pages. We can also set the same properties to the all controls in the web application. Theme is the centralized code which we can use anywhere in our web application. **Theme includes the two types of files: CSS file and skin file in the theme folder.**

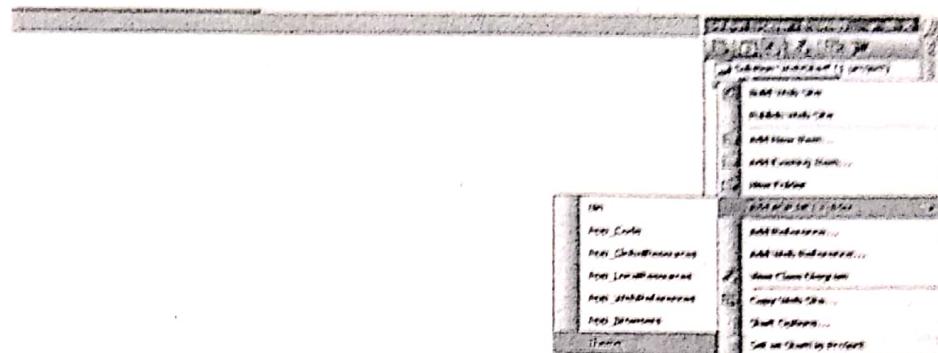
CSS file is used to create same look across the web application. This file is used to set same background, font, border, etc. This CSS file is an external CSS which we must not include the head tag while we are using theme. Because this CSS file is in the Theme folder. So we just have to include the Theme file in the page.

Skin file is used to set the same properties for the all controls. If we set the properties of the all controls in this file, then we don't require setting the properties individually to the controls in the webpage. We just have to include the Theme file in the page.

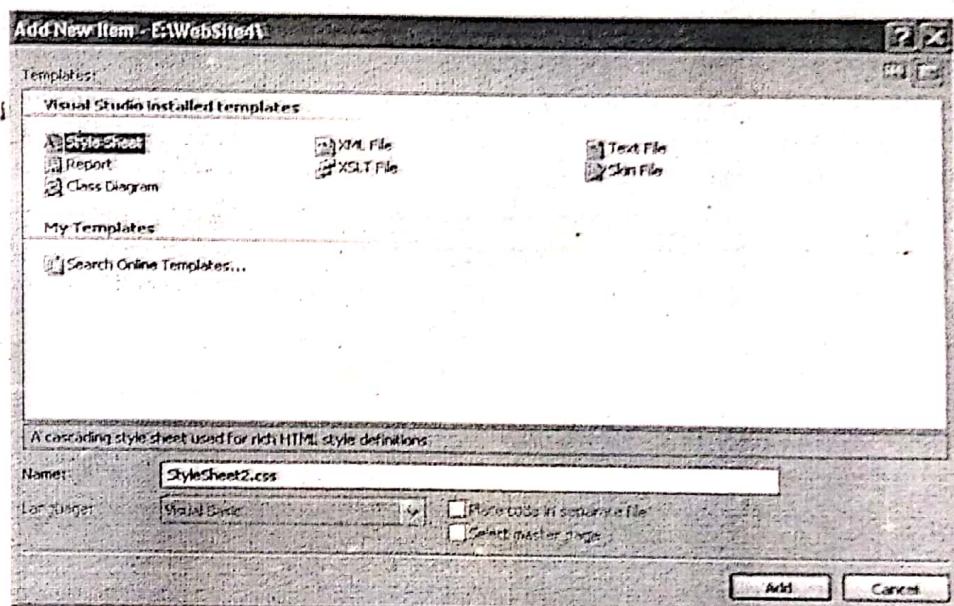
- How to create Theme?

Steps to create theme:

1. Right click on our application in the solution explorer and click on add asp.net folder and in that click on Theme.



2. Now right click on the theme1 from there you can add CSS file as well Skin file in the folder.



3. You can code for the CSS file same like external CSS. While using the theme, we don't need to add this file in the head tag. The effect of the css file, we can see run time.
4. To add the code in the skin file:
 - a. Go to the web form
 - b. Add the control in the web form which you want to put in the skin file.
 - c. Set the properties on that control in the web form.
 - d. Copy the source code of that control and paste in the skin file.

e. Remove the id and text property of that control from the skin file. Because these properties cannot be the same for all controls.

5. You can create the multiple themes in the Theme folder.

- How to use Theme?

Steps to add the theme in the file statically:

1. Open the web page in which you want to put theme.
2. Open the source part of that web page.
3. Go to the first line of that web page in the source part.
4. Go to the end of the line and type Theme="Theme1"

Steps to add the theme in the file Dynamic:

1. Go to the coding part which is the aspx.vb
2. Add the pre Init event of the page.
3. In that write code Theme="Theme1"

You can see the effect of the theme runtime. So you have to run the web application.

Static implementation of Theme:

```
<%@ Page Language="VB" AutoEventWireup="true" CodeFile="Default.aspx.vb" Inherits="_Default" Theme="Theme1" %>
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

Dynamic code:

```
Protected Sub Page_PreInit(ByVal sender As Object, ByVal e As Sy
    Theme = "Theme1"
End Sub
```

2. HTML Controls:

The HTML controls in the web page can be used client side. So we can use these controls when we don't need server side coding. To show the output of the control we need to use the client scripting language. The client scripting languages are javascript and vbscript. When we don't require any database connection and just do some application like calculator, games in that we should use the HTML controls.

These controls are light weighted in comparison of asp.net controls. So we should use these controls when we don't require any server side coding. These controls provide fastest output to the client because the coding is at client side. These controls don't provide any server side functionality or security because coding is at client side.

We can use different HTML controls like:

Input(Button)	Simple html control to raise the client side event.
Input(Submit)	Submit button used to process the data client side which is submitted from the web page.
Input(Restore)	To clear the data of the web page.
Input(Text)	To show the text box.
Input(File)	To show the file upload control in the web page.
Input>Password	To use the password character in the textbox in the web page.
Input(Checkbox)	For multiple selection and show the checkbox in the web page.
Input(Radio)	For single selection and show the radio button in the web page.
Input(Hidden)	To store the value hidden in the web page.
TextArea	To show the multi line text box in the web page.
Table	To use the table control in the web page.

Image	To use the image in the web page.
Select	To use the drop down control in the web page.
Horizontal Rule	To show the horizontal line in the web page.
Div	This control can be use in place of table control in the web page.

We can use the different client side events like:

- onclick
- ondblclick
- onkeydown
- onkeypress
- onkeyup
- onmousedown
- onmouseup
- onmousemove
- onmouseout
- onmouseover
- onmouseup

9. Ad Rotator:

This control is used to display different advertisements as per value. The advertisements will be displayed randomly. The advertisement will be changed after refresh the webpage or reopen the website.

Code of xml file:

```
<?xml version="1.0" encoding="utf-8" ?>
<Advertisements>
    <Ad>
        <ImageUrl>Images/1.jpg</ImageUrl>
        <NaviagateUrl>www.mkbhavuni.edu.in</NaviagateUrl>
        <AlternateText>TYBCA RESULT DECLARED...</AlternateText>
        <Impressions>500</Impressions>
        <Keyword>University</Keyword>
    </Ad>
</Advertisements>
```

Note:

1. All tags of xml file is case sensitive.
2. Write all advertisements code into xml file and set the property of adrotator is AdvertisementFile.(Compulsory)

Description of different tags:

Advertisements	This tag is collection of different advertisements. The xml file code start from this tag.
Ad	Each advertisement starts with Ad tag. If we want to display 5 advertisement then we have to use 5 Ad tag.
ImageUrl	We have to mention the path of the image(advertisement). For e.g. <code><ImageUrl>Images/1.jpg</ImageUrl></code>
NavigateUrl	We have to mention the link to which we have to redirect the page. For e.g.: <code><NavigateUrl>www.google.com</NavigateUrl></code>