Chapter 1

An introduction to ASP.NET programming
The components of a web application

- Web Server
- Computer
- Tablet
- Smart Phone
- The Internet
A static web page
How a web server processes a static web page

Web Browser

HTTP request

HTTP response
(with HTML)

Web Server
(HTML files)
A dynamic web page

Halloween Superstore
For the little goblin in all of us

Please select a product  Michael Head

**Michael Head**

Mini Michael Meyers head
For classic horror lovers! The infamous Halloween murderer’s shrunken head is a perfect party prop.

$29.99

Quantity

Add to Cart  Go to Cart
How a web server processes a dynamic web page

HTTP request
HTTP response (with HTML)

Web Browser
Web Server (IIS)
Application Server (ASP.NET)
Database Server

Web Browser

Murach's ASP.NET 4.5/C#, C1
© 2013, Mike Murach & Associates, Inc.
The two main ASP.NET technologies

- ASP.NET Web Forms
- ASP.NET MVC

Three other ASP.NET technologies

- ASP.NET Web Pages with Razor
- ASP.NET Dynamic Data Entities
- ASP.NET Reports Web Site
The four editions of Visual Studio 2012

- Visual Studio Express 2012 for Web
- Visual Studio Professional 2012
- Visual Studio Premium 2012
- Visual Studio Ultimate 2012

The free edition of Visual Studio for web apps

- Visual Studio 2012 Express for Web

What the free edition includes

- IIS Express
- SQL Server Express LocalDB
Standalone development

Windows 7 or later
.NET Framework 4.5
Visual Studio 2012
IIS Express
SQL Server

What you need on your PC

- Windows 7 or later
- .NET Framework 4.5
- Visual Studio 2012
- IIS Express
- SQL Server Express LocalDB
Why state is difficult to track in a web application

First HTTP request: The browser requests a page.

First HTTP response: The server returns the requested page and the application ends.

Next HTTP request: The browser requests another page. The server has no way to associate the browser with its previous request.
Five ASP.NET features for maintaining state

- View state (chapter 2 and 8)
- Session state (chapter 4 and 8)
- Application state (chapter 8)
- Server-side caching (chapter 8)
- Profiles
The Future Value application after the user clicks the Calculate button
The Future Value application with error messages displayed

Interest rate must range from 1 to 20. Number of years is required.
The Future Value form in Design view of Visual Studio 2012

401K Future Value Calculator

Monthly investment: Unbound
Annual interest rate: 6.0
Number of years: 10
Future value: [lblFutureValue]

Calculate  Clear

Interest rate is required. Interest rate must range from 1 to 20. Number of years is required. Years must range from 1 to 45.
The files in the Future Value application

<table>
<thead>
<tr>
<th>Folder</th>
<th>File</th>
</tr>
</thead>
<tbody>
<tr>
<td>(root)</td>
<td>Default.aspx</td>
</tr>
<tr>
<td>(root)</td>
<td>Default.aspx.cs</td>
</tr>
<tr>
<td>(root)</td>
<td>web.config</td>
</tr>
<tr>
<td>Images</td>
<td>MurachLogo.jpg</td>
</tr>
</tbody>
</table>
The aspx file for the Default form (Default.aspx)

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

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head id="Head1" runat="server">
    <title>Chapter 1: Future Value</title>
    <style><!-- CSS code for the generated styles -->
</style>
</head>
<body>
<form id="form1" runat="server">
    <div>
        <img alt="Murach" class="style1" src="Images/MurachLogo.jpg" />
        <h1 style="color: #0000FF">401K Future Value Calculator</h1>
        <table class="style2">
            <tr>
                <td class="style3">Monthly investment</td>
                <td><asp:DropDownList ID="ddlMonthlyInvestment" runat="server" Width="106px"></asp:DropDownList></td>
            </tr>
            <tr>
                <td class="style3">Annual interest rate</td>
                <td><asp:TextBox ID="txtInterestRate" runat="server" Width="100px">6.0</asp:TextBox></td>
            </tr>
            <tr>
                <td class="style3">Number of years</td>
                <td><asp:TextBox ID="txtNumberOfYears" runat="server" Width="100px"></asp:TextBox></td>
            </tr>
        </table>
    </div>
</form>
</body>
</html>
```
The aspx file for the Default form (cont.)

```xml
<tr>
    <td><asp:TextBox ID="txtYears" runat="server" Width="100px">10</asp:TextBox></td>
</tr>
<tr>
    <td class="style3">Future value</td>
    <td><asp:Label ID="lblFutureValue" runat="server" Font-Bold="True"></asp:Label></td>
</tr>
<tr>
    <td class="style3">&nbsp;</td>
    <td>&nbsp;</td>
</tr>
<tr>
    <td class="style3"><asp:Button ID="btnCalculate" runat="server" Text="Calculate" Width="100px"
         OnClick="btnCalculate_Click" /></td>
    <td><asp:Button ID="btnClear" runat="server" Text="Clear" Width="100px" OnClick="btnClear_Click"
         CausesValidation="false" /></td>
</tr>
<br />
<!--[aspx code for the field validators -->
</div>
</form>
</body>
</html>
```
The code-behind file for the Default form (Default.aspx.cs)

```csharp
using System;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            for (int i = 50; i <= 500; i += 50)
                ddlMonthlyInvestment.Items.Add(i.ToString());
        }
    }
}
```
The code-behind file for the Default form (cont.)

```csharp
protected void btnCalculate_Click(object sender, EventArgs e)
{
    if (IsValid)
    {
        int monthlyInvestment = Convert.ToInt32(ddlMonthlyInvestment.SelectedValue);
        decimal yearlyInterestRate = Convert.ToDecimal(txtInterestRate.Text);
        int years = Convert.ToInt32(txtYears.Text);

        decimal futureValue = this.CalculateFutureValue(monthlyInvestment, yearlyInterestRate, years);

        lblFutureValue.Text = futureValue.ToString("c");
    }
}
```
protected decimal CalculateFutureValue(int monthlyInvestment,
    decimal yearlyInterestRate, int years)
{
    int months = years * 12;
    decimal monthlyInterestRate = yearlyInterestRate / 12 / 100;
    decimal futureValue = 0;

    for (int i = 0; i < months; i++)
    {
        futureValue = (futureValue + monthlyInvestment)
            * (1 + monthlyInterestRate);
    }
    return futureValue;
}

protected void btnClear_Click(object sender, EventArgs e)
{
    ddlMonthlyInvestment.SelectedIndex = 0;
    txtInterestRate.Text = "";
    txtYears.Text = "";
    lblFutureValue.Text = "";
}
}