

2022/2023(2)

IF184504 Web Programming

Lecture #9

**Dynamic ASP.NET Page, Web Form &
Web Control**

Misbakhul Munir **IRFAN SUBAKTI**

司馬伊凡

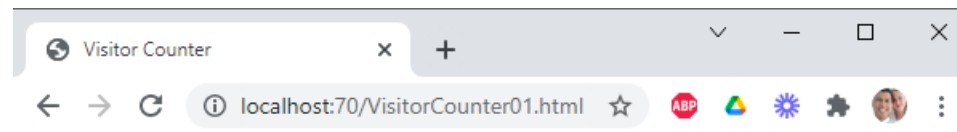
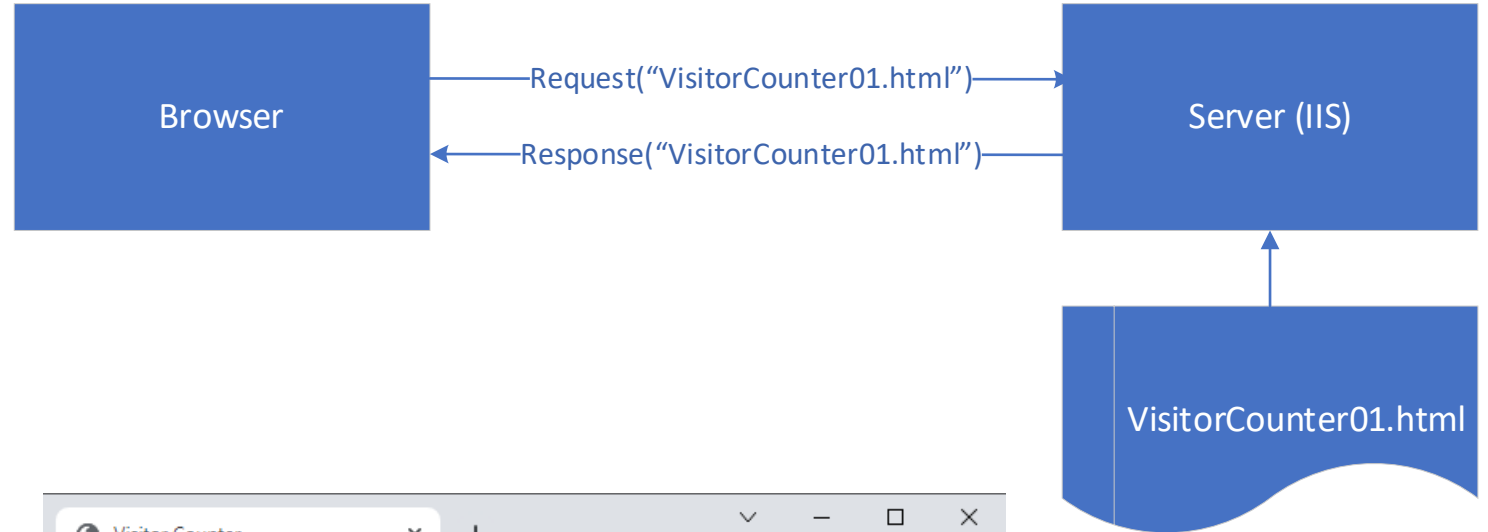
Мисбакхул Мунир **Ирфан Субакти**

Web pages: Category

- Static page
 - E.g., any HTML page
- Dynamic page
 - E.g., `.aspx` page (ASP.NET)

Static Page: VisitorCounter01.html

```
VisitorCounter01.html  x
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <meta charset="utf-8" />
5    <title>Visitor Counter</title>
6  </head>
7  <body>
8    <h1>Welcome</h1>
9    You are visitor number 1!
10 </body>
11 </html>
```

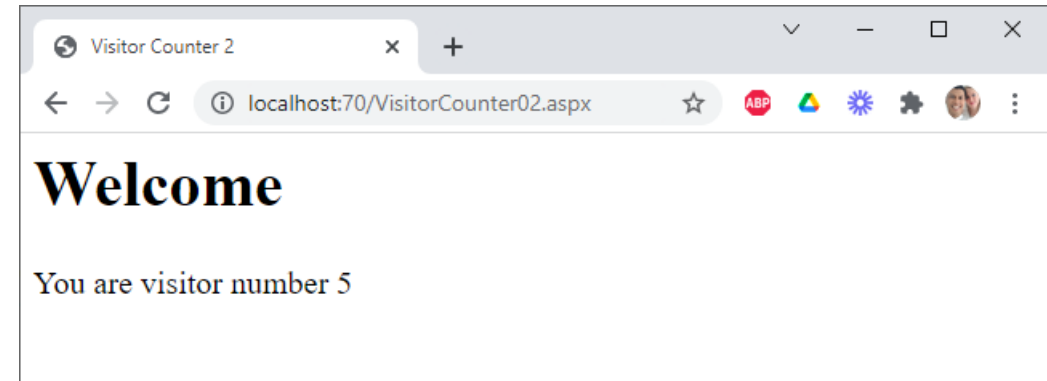


Welcome

You are visitor number 1!

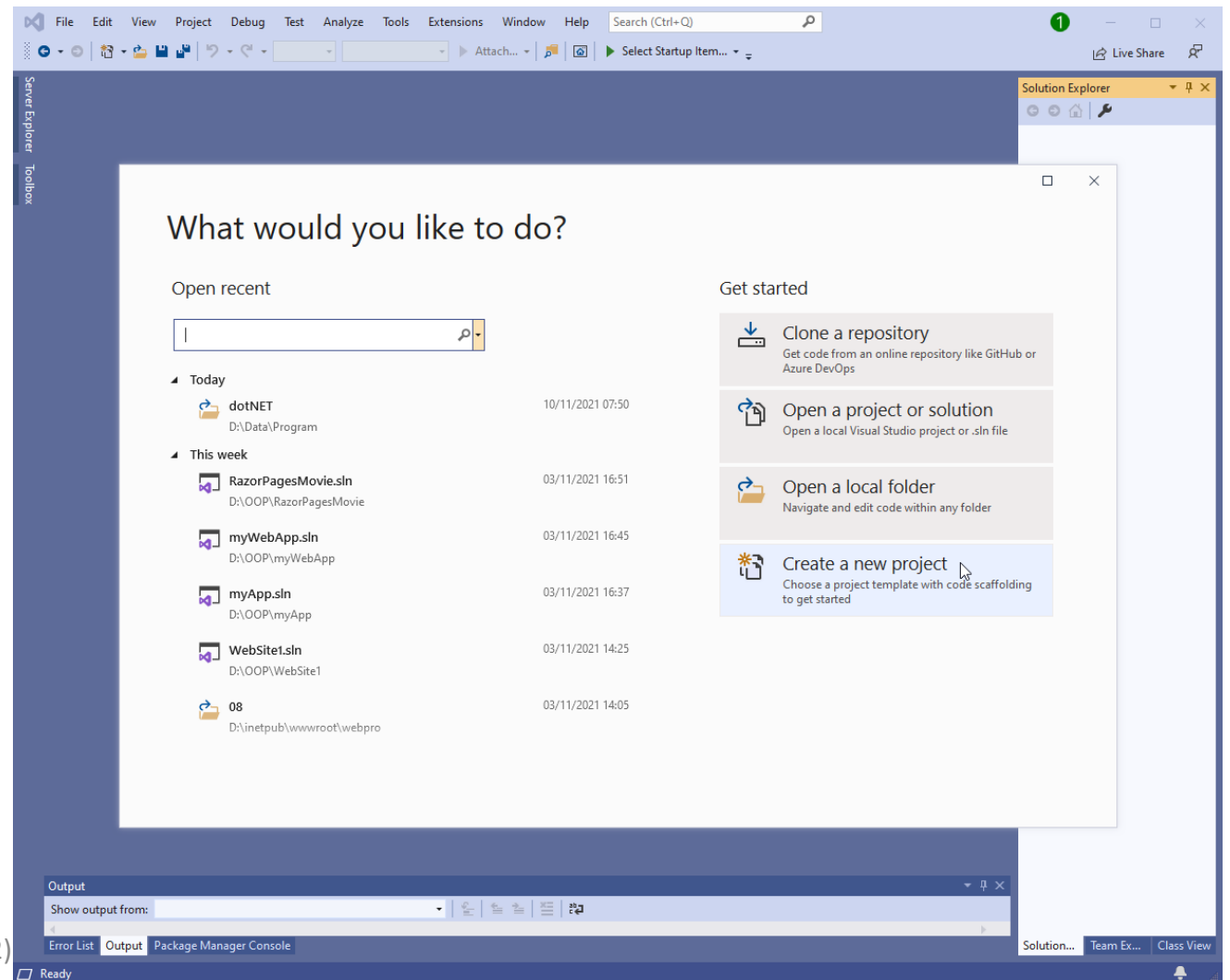
Dynamic Page: VisitorCounter02.aspx

```
VisitorCounter02.aspx
1 <%@ Page Language="C#" %>
2 <%@ Import Namespace="System.IO" %>
3 <html>
4 <head>
5 <title>Visitor Counter 2</title>
6 </head>
7 <body>
8 <h1>Welcome</h1>
9 You are visitor number
10 <%
11     FileStream s = new FileStream("D:/ITS/2021 ITS/08 Web Programming/.NET/Lecture9/VisitorCounter/" +
12         "VisitorCounter.dat", FileMode.OpenOrCreate);
13     int n;
14     try {
15         BinaryReader r = new BinaryReader(s);
16         n = r.ReadInt32();
17     } catch {
18         n = 0;
19     }
20     n++;
21     s.Seek(0, SeekOrigin.Begin);
22     BinaryWriter w = new BinaryWriter(s);
23     w.Write(n);
24     s.Close();
25     Response.Write(n);
26 %>
27 </body>
28 </html>
```



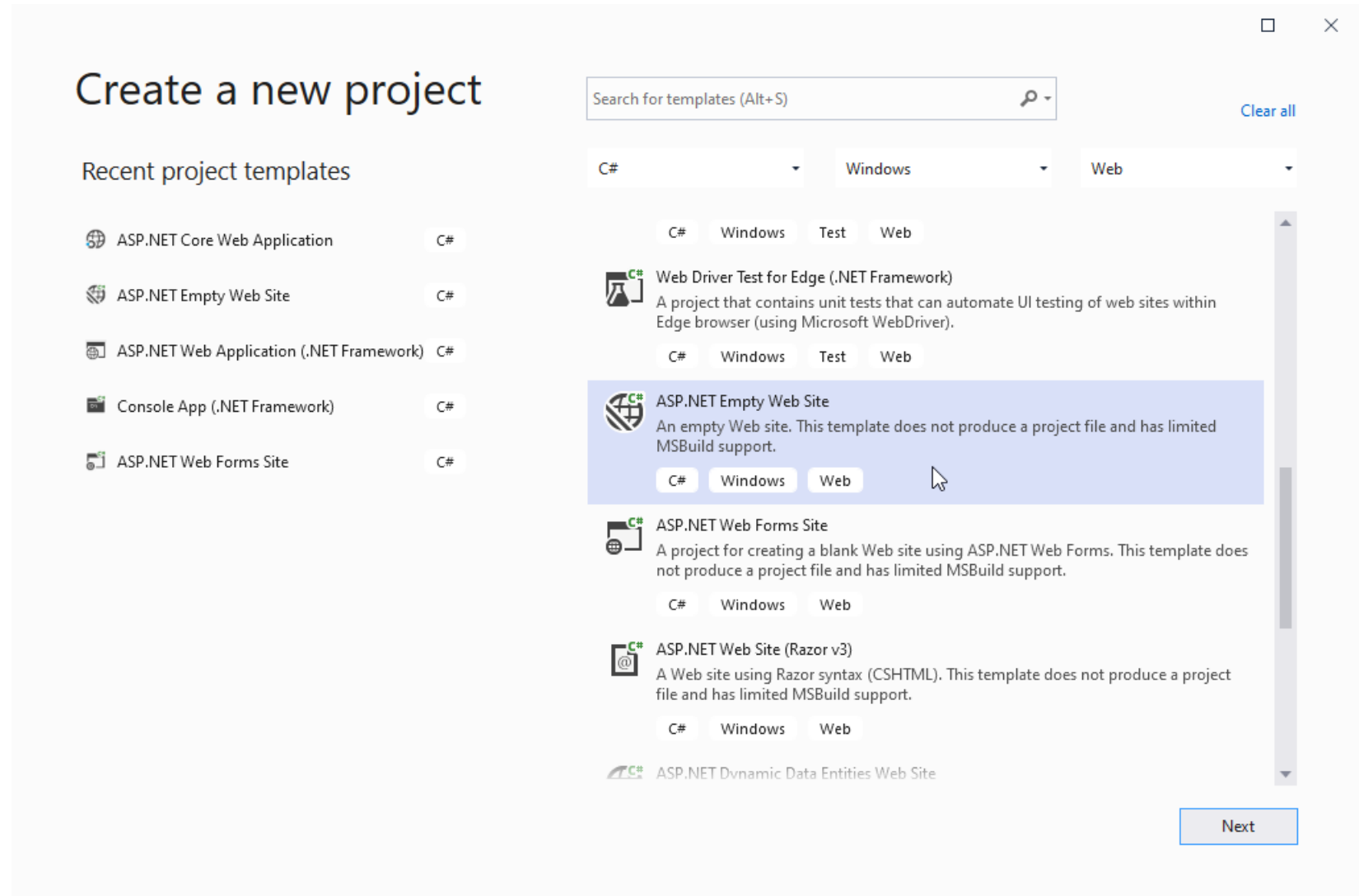
Visual Studio 2019: Creating a new project

- Open Visual Studio 2019 (VS2019)
- Choose/click **Create a new project**



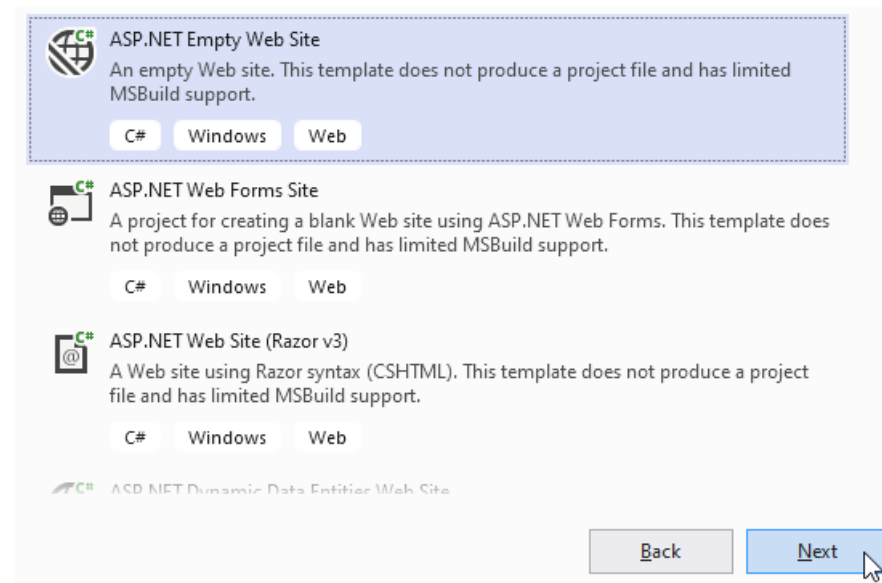
VS2019: Choose a project template

- Choose **ASP.NET Empty Web Site**



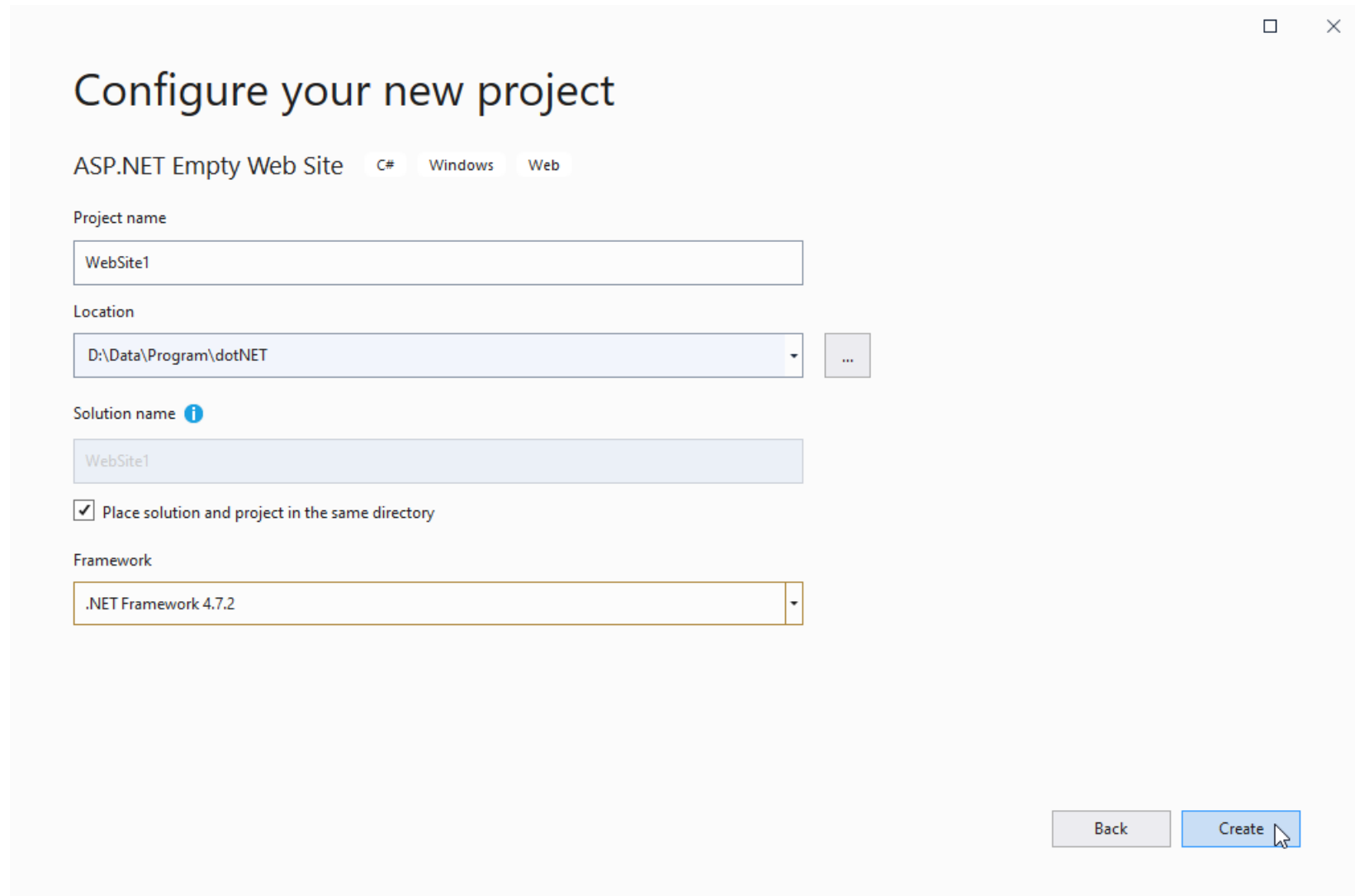
VS2019: Choose a project template (cont'd)

- Click **Next**



VS2019: Choose a project's name & folder

- Click **Create**



Configure your new project

ASP.NET Empty Web Site C# Windows Web

Project name

WebSite1

Location

D:\Data\Program\dotNET

Solution name ⓘ

WebSite1

Place solution and project in the same directory

Framework

.NET Framework 4.7.2

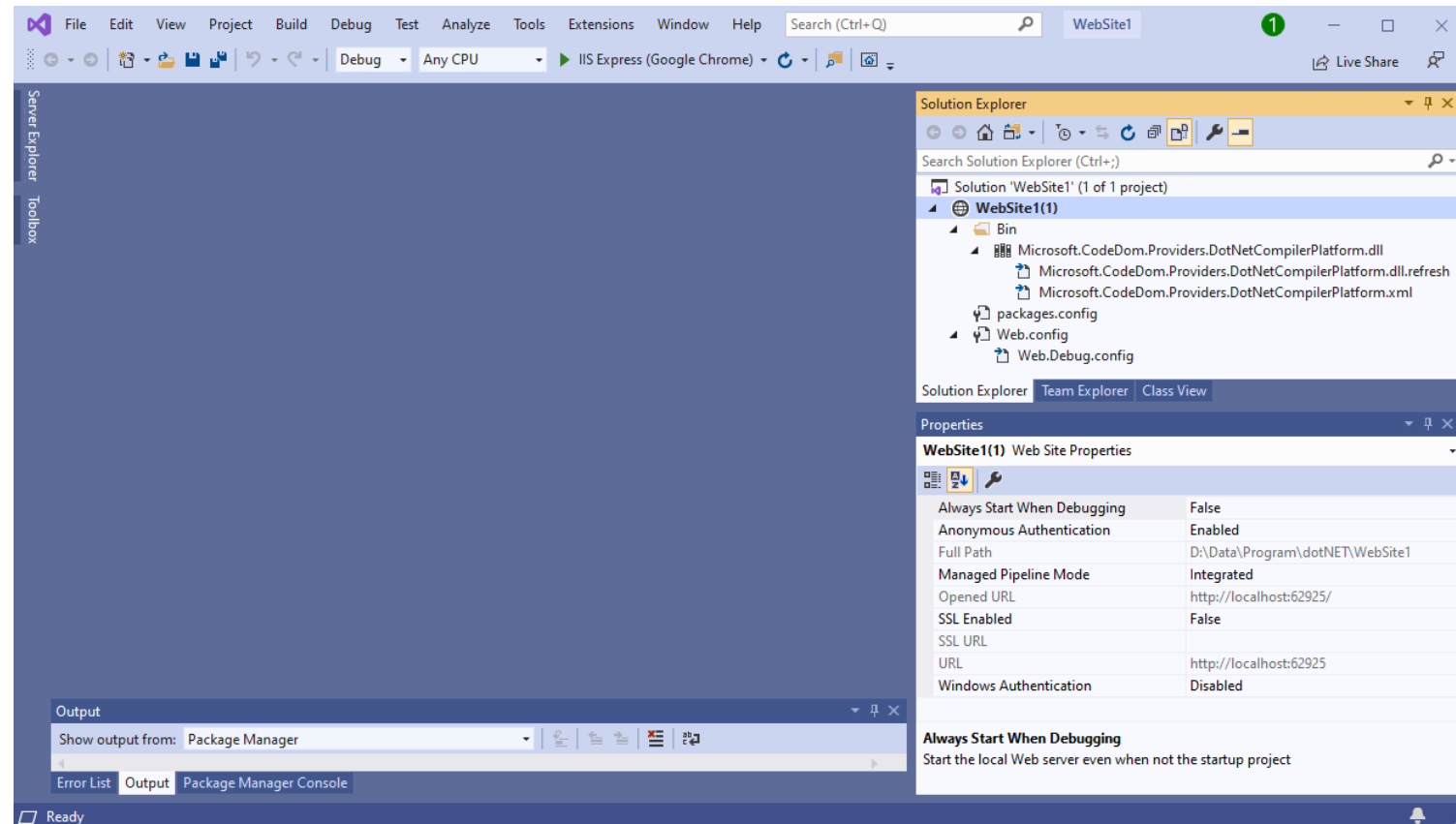
Back Create

VS2019: Creating project

- It takes some times for VS2019 to create all the necessary files

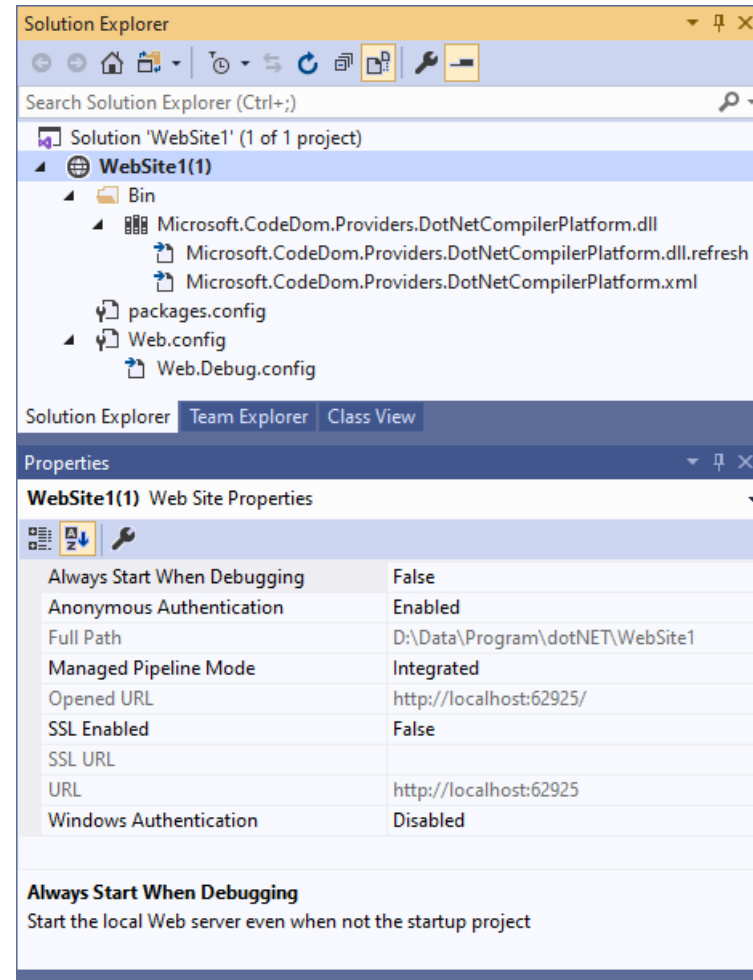


- Once it finished, it shows:



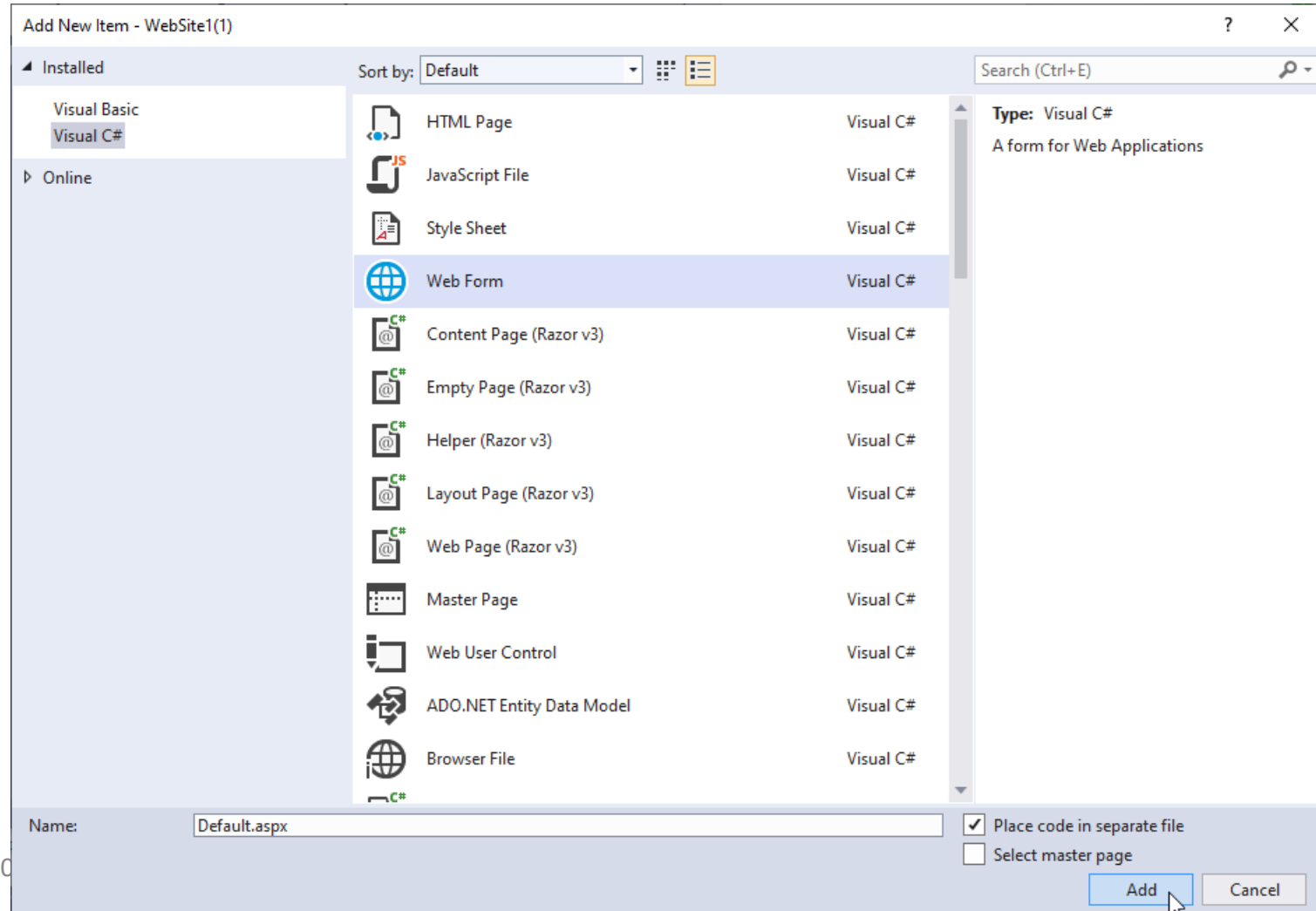
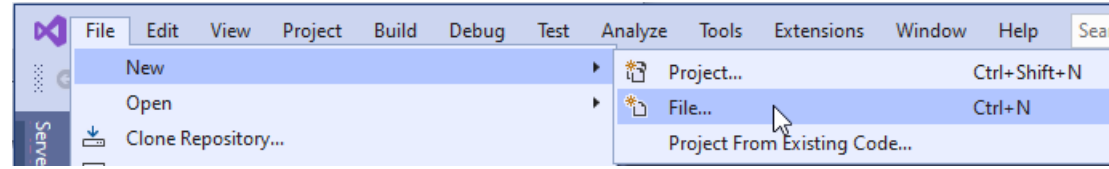
VS2019: Solution Explorer & Properties

- VS2019 shows **Solution Explorer** and **Properties** windows



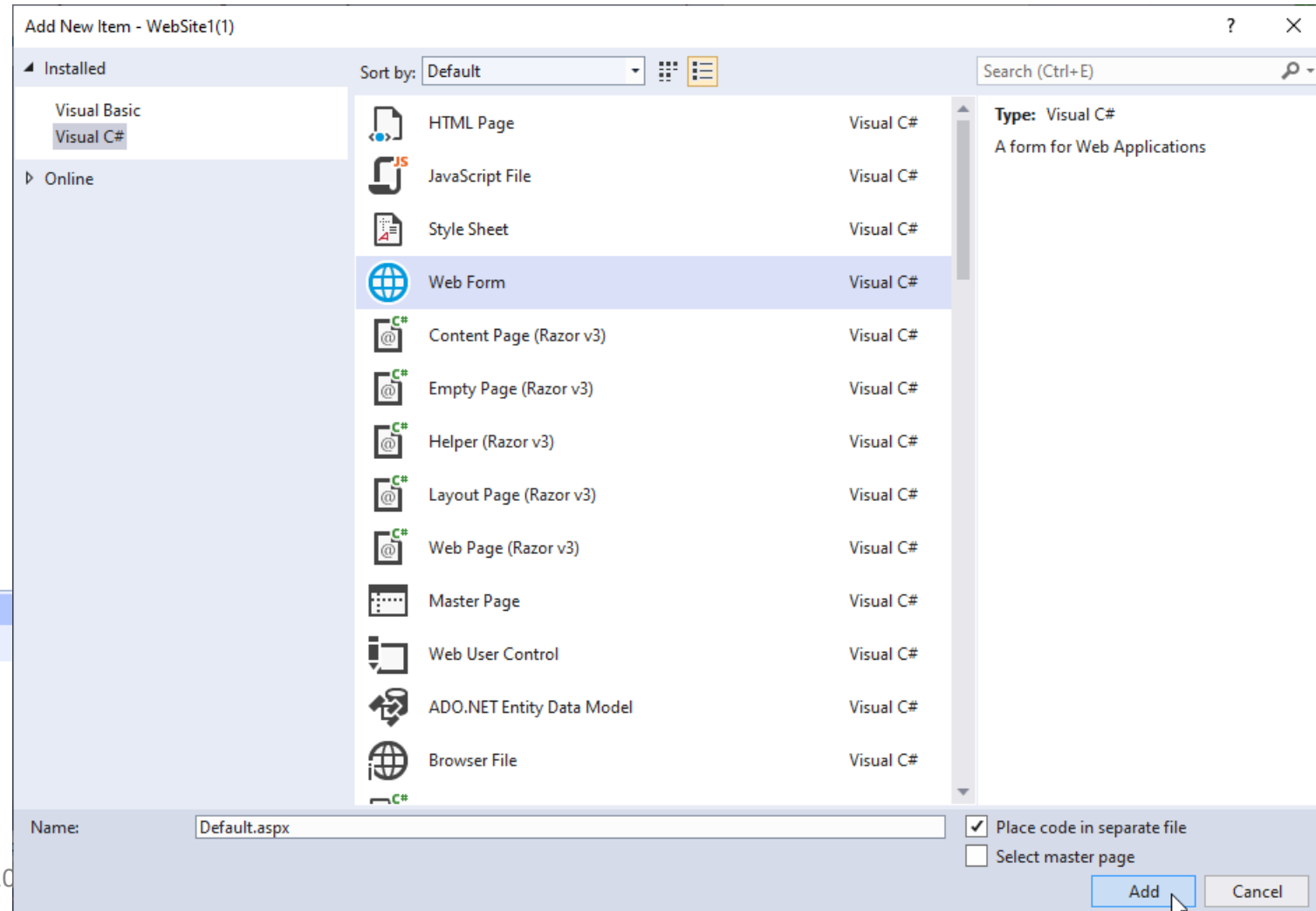
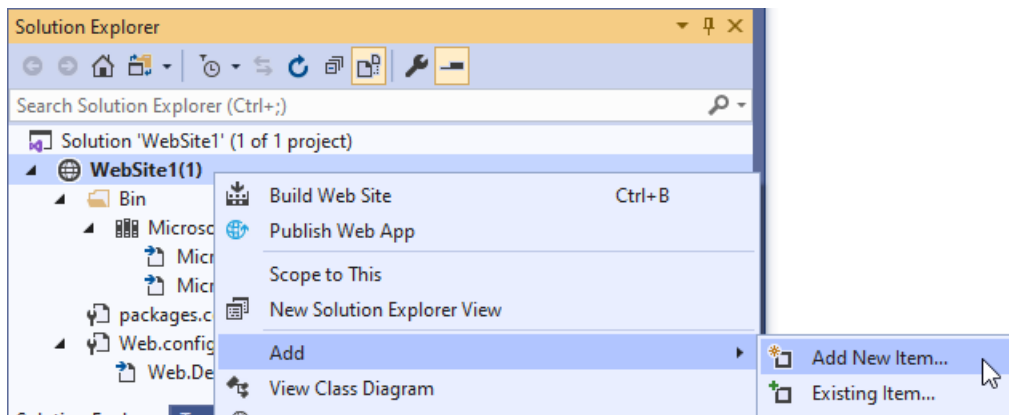
VS2019: New File

- Click **File > New > File...**
- Choose **Web Form**



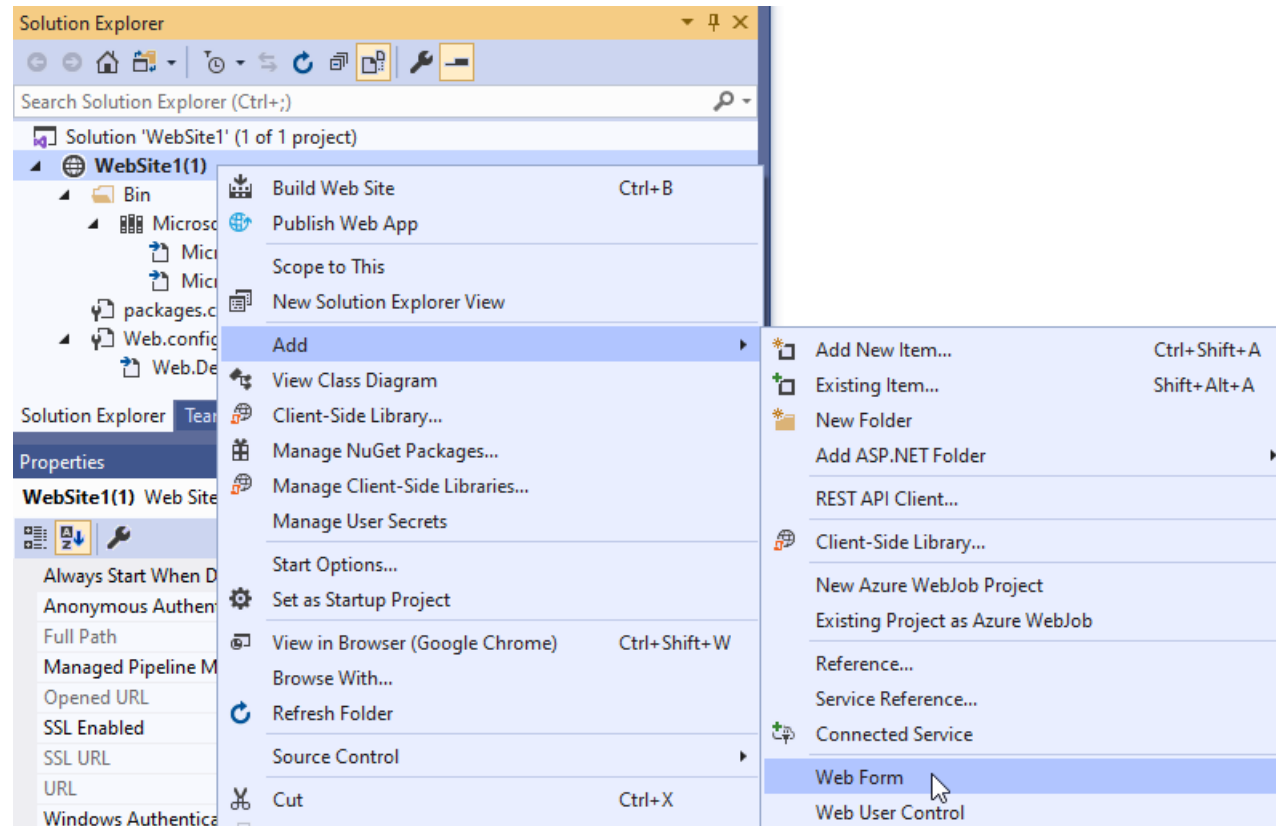
VS2019: New File (continued)

- ... or you may do this:
right click on the **project name** at **Solution Explorer**



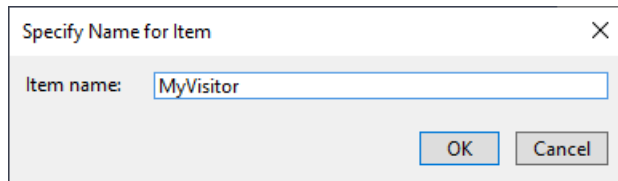
VS2019: New File (continued)

- If an item is frequently chosen then it will come up right away after right click on **project name** at **Solution Explorer** > **Add** > [freq. item]

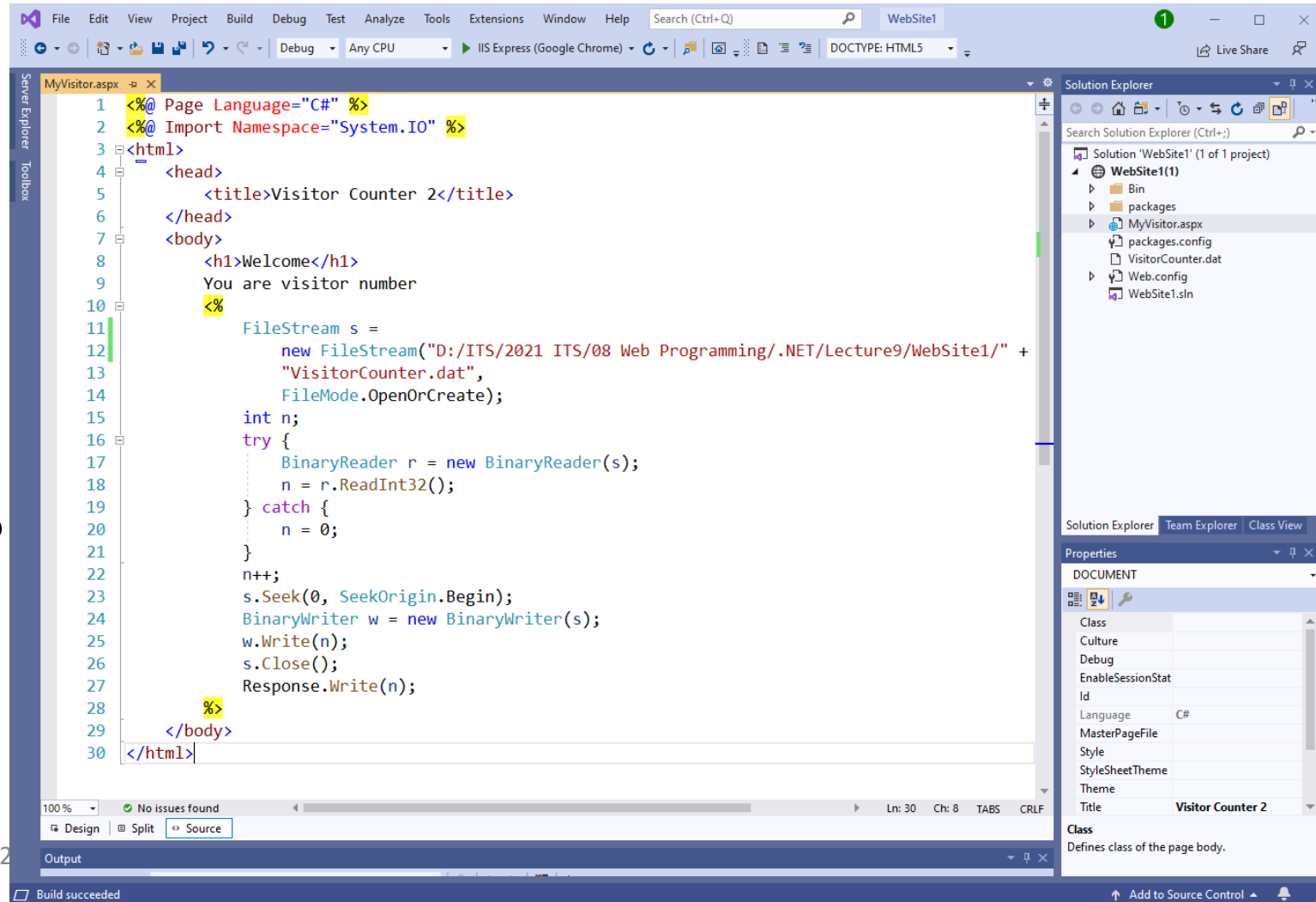


VS2019: Specify Name for Item

- Write down a new item's name



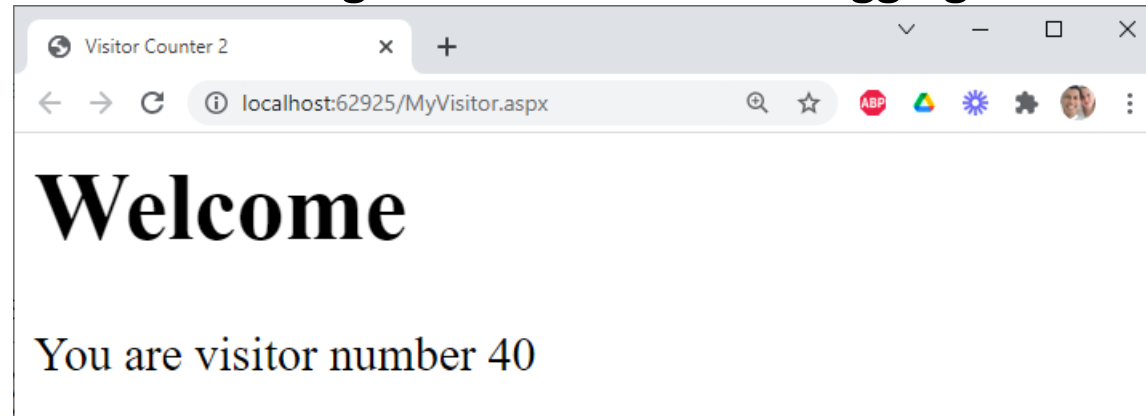
- Copy the previous file's content
VisitorCounter02.aspx
x to the newly created file
MyVisitor.aspx



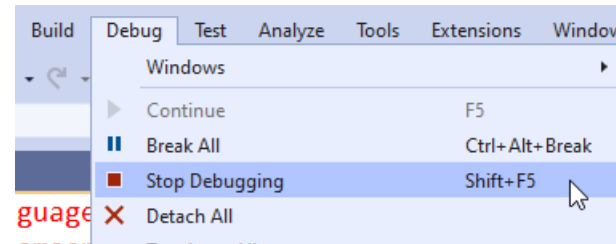
VS2019: Start (Without) Debugging

- To run our project
 - Press **F5** or click **Debug > Start Debugging**, OR
 - Press **Ctrl+F5** or click **Debug > Start Without Debugging**

- Debugging:



- To stop debugging:

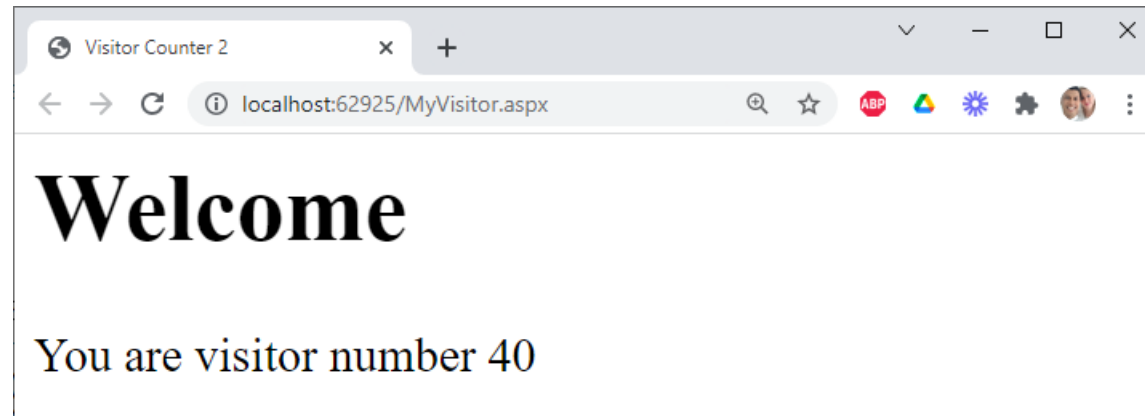


- ... then the window's result will be exited/stopped

VS2019: Start (Without) Debugging

- To run our project
 - Press **F5** or click **Debug > Start Debugging**, OR
 - Press **Ctrl+F5** or click **Debug > Start Without Debugging**

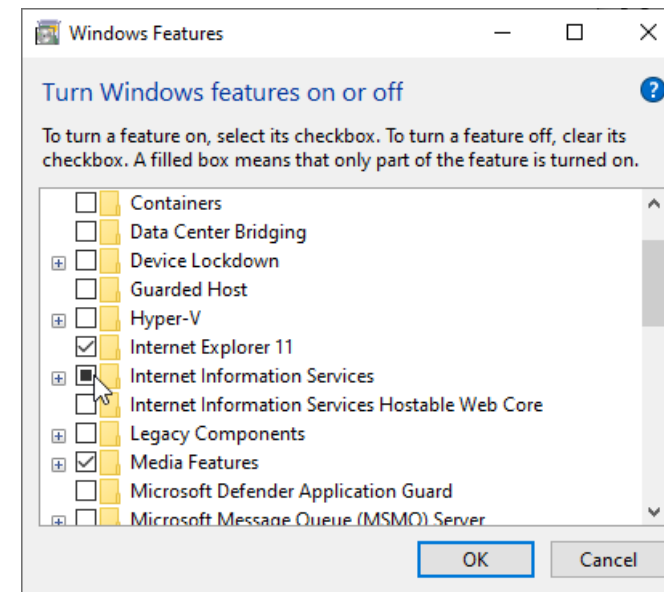
- Without Debugging:



- To stop: just exit/close the window's result above

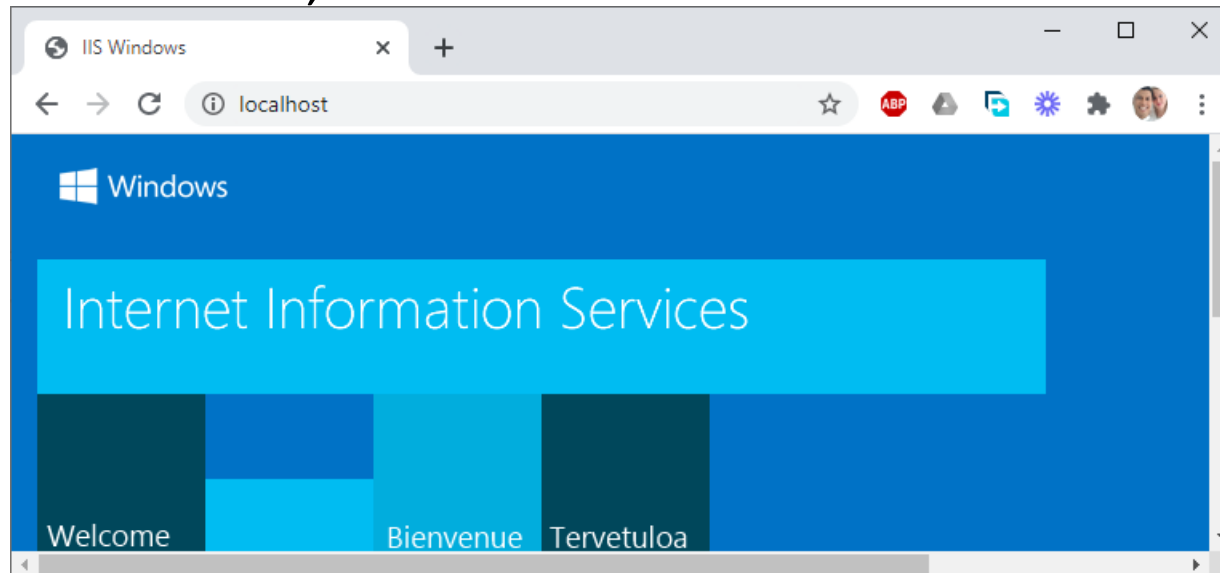
Internet Information Service (IIS): Requirement for our project

- IIS need to be activated to run our VS2019 projects.
 - Menu > Control Panel > Program > Turn Windows features on or off > Windows Features > Internet Information Services > Check it!
 - Once it's checked, IIS will be activated!



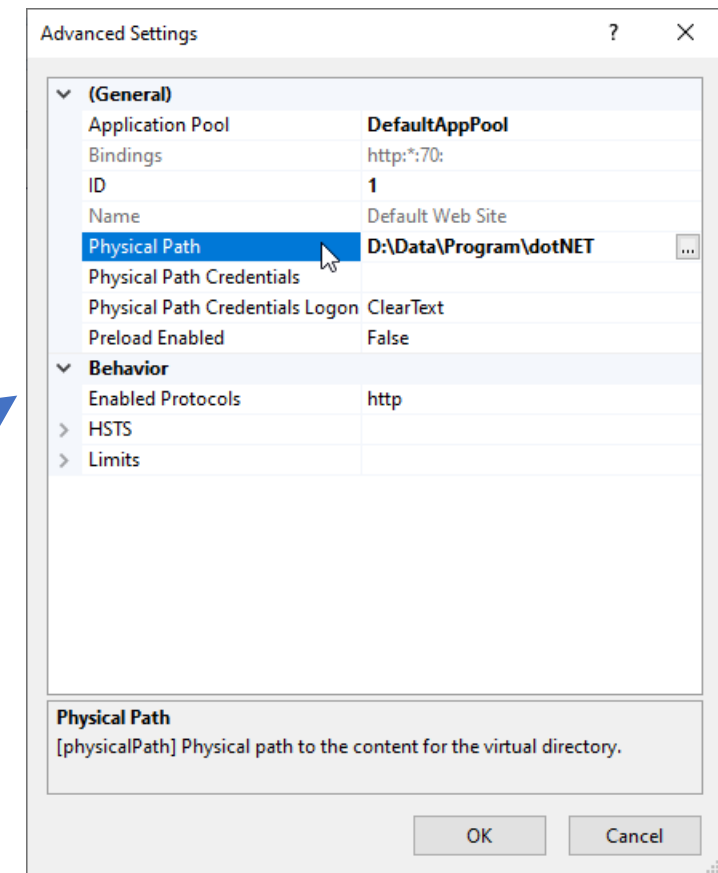
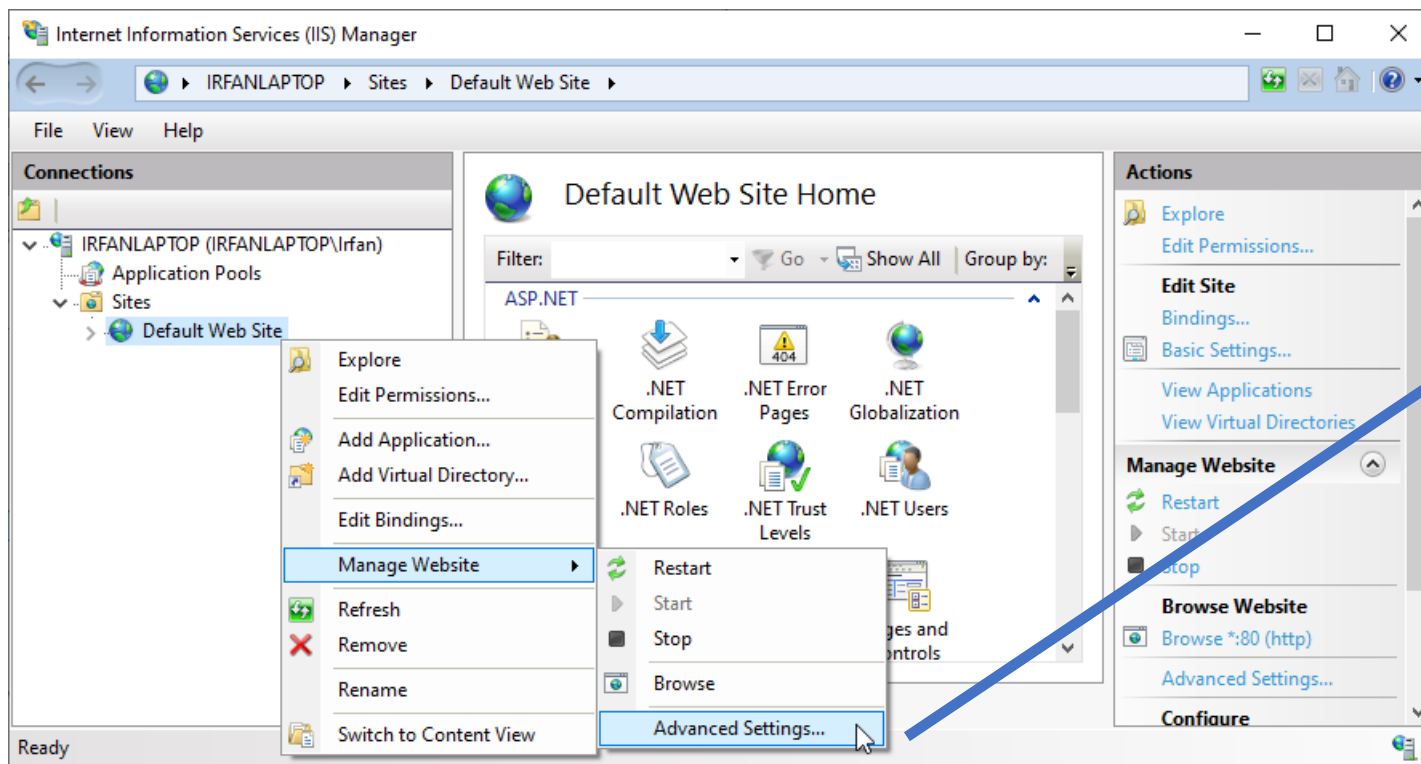
IIS: Working check

- Check whether IIS works
 - Open your internet browser
 - Type `localhost` (if we have set a particular port, e.g., port 70, then it would be `localhost:70`) on address bar
 - If you can see this, it's mean IIS works!



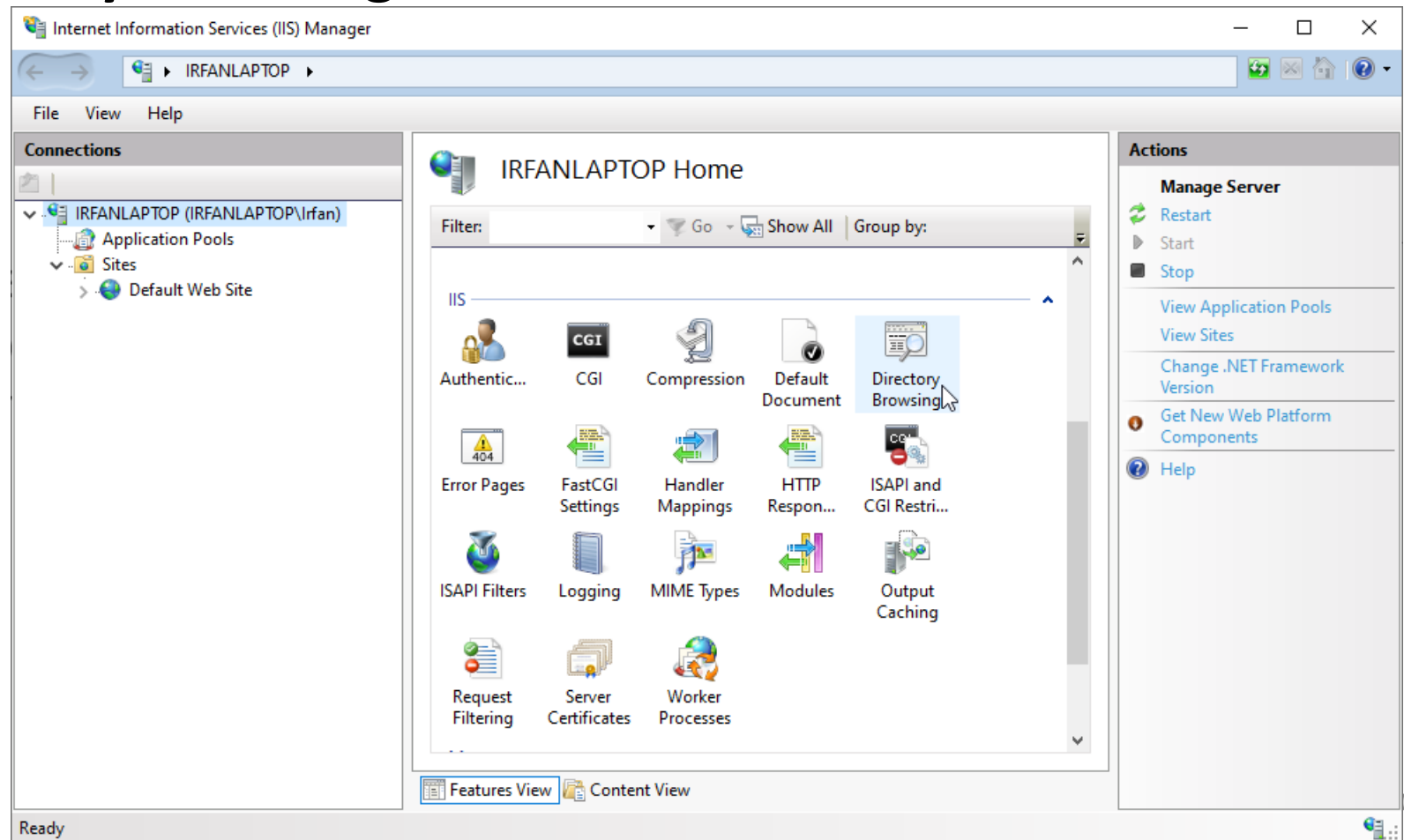
IIS: Working folder

- Choose the folder for our *working folder*, e.g.,
D:\Data\Program\dotNET



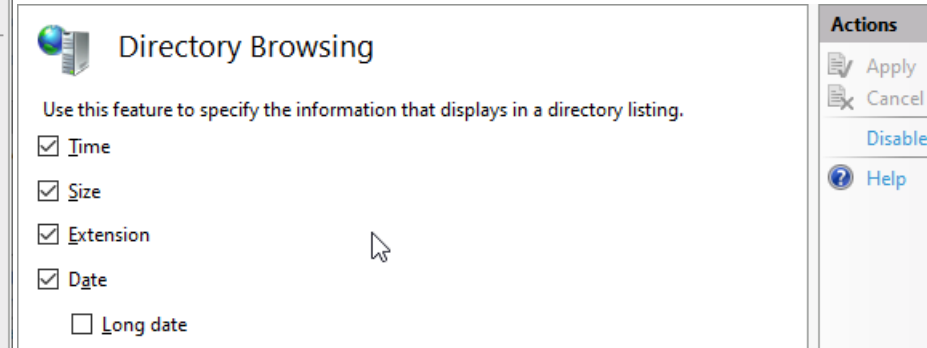
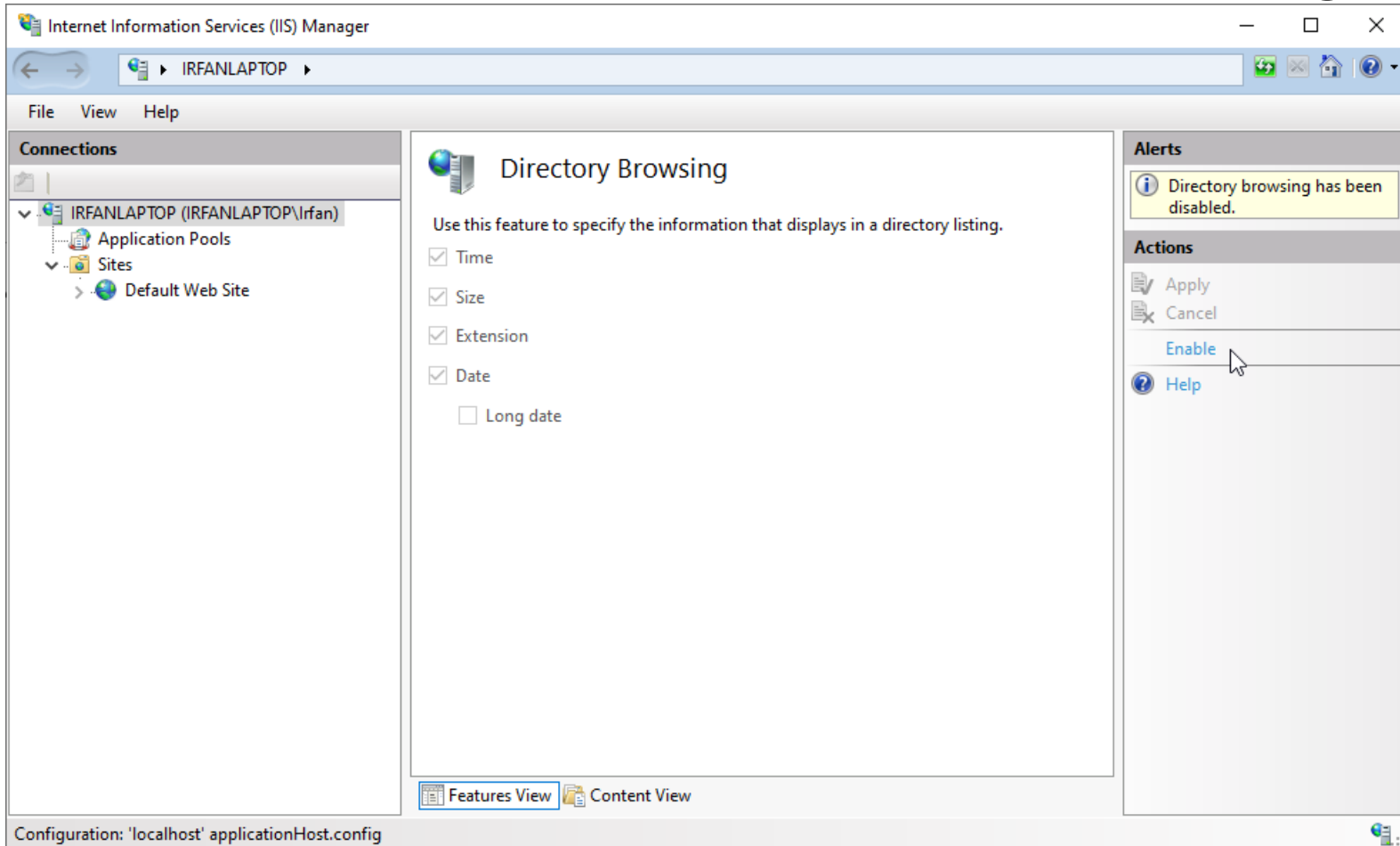
IIS: Directory Browsing

- Double click **Directory Browsing**



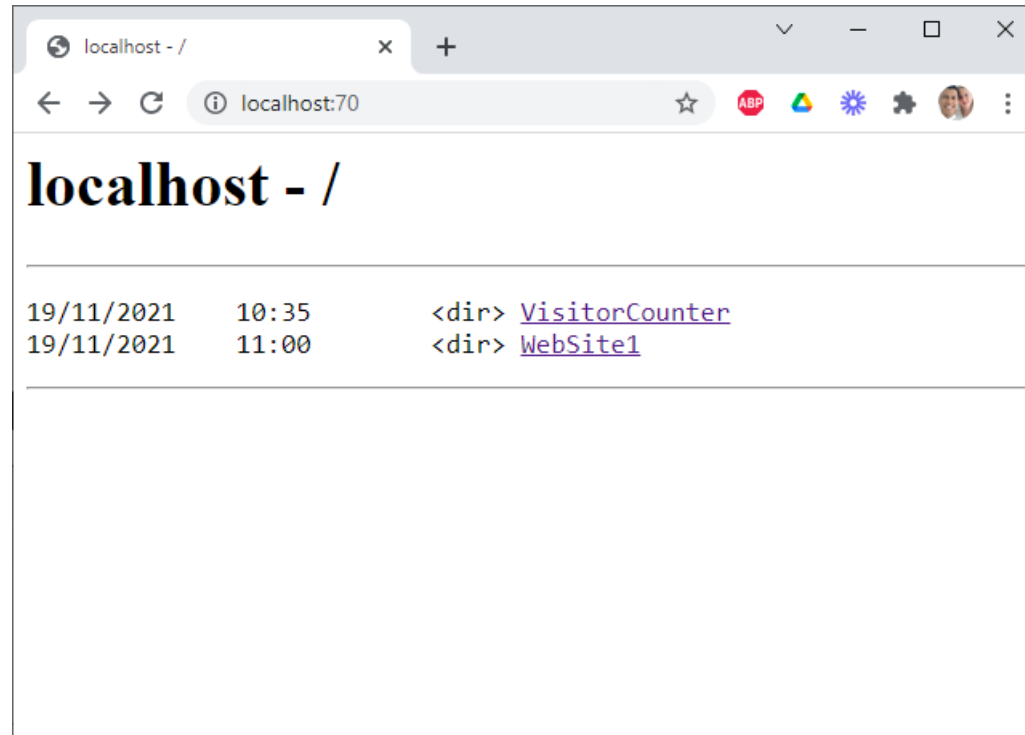
IIS: Directory Browsing (continued)

- Click **Enable** → we can browse our working folder



IIS: Directory Browsing enable

- Now we can browse our working folder



WebSite2: Script code

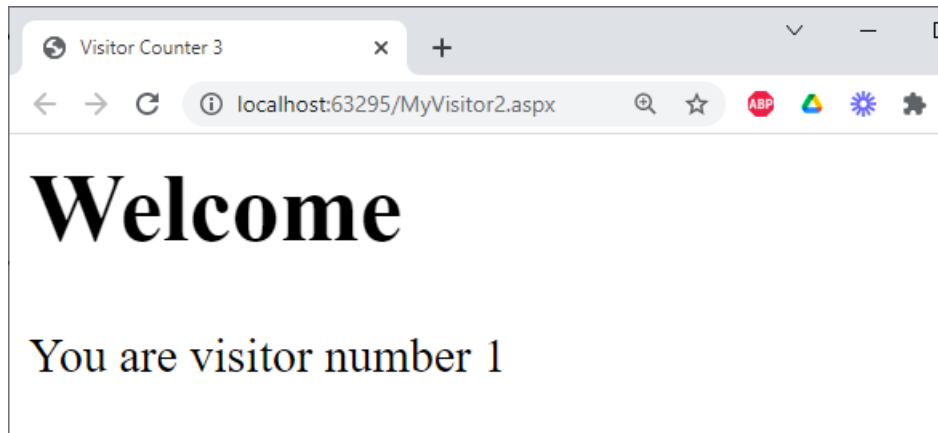
- Create a new project, WebSite2, using a similar process as WebSite1
- Make a file `MyVisitor2.aspx` like this:

```
MyVisitor2.aspx  ↵ ×
1  <%@Page Language="C#" Inherits="MyVisitor2" Src="MyVisitor2.aspx.cs"%>
2  <html>
3  <head>
4      <title>Visitor Counter 3</title>
5  </head>
6  <body>
7      <h1>Welcome</h1>
8      You are visitor number <%=MyCounter()%>
9  </body>
10 </html>
```

- File `MyVisitor2.aspx` has Inherits from class `MyVisitor2` which can be seen in file `MyVisitor2.aspx.cs`

WebSite2: Script code (continued)

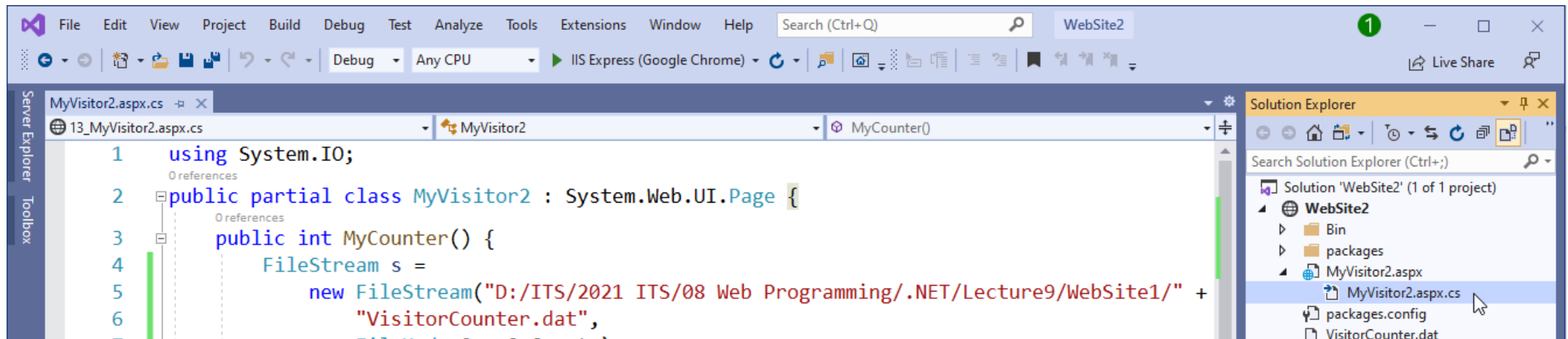
- File `MyVisitor2.aspx.cs` has a program to calculate the number of visitor like our previous project



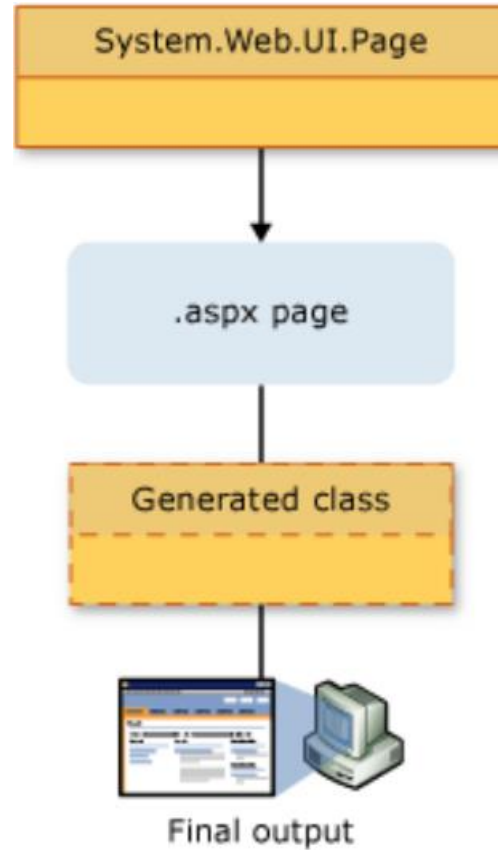
```
MyVisitor2.aspx.cs -> X
13_MyVisitor2.aspx.cs -> MyVisitor2 -> MyCounter()
1 using System.IO;
2 public partial class MyVisitor2 : System.Web.UI.Page {
3     public int MyCounter() {
4         FileStream s =
5             new FileStream("D:/ITS/2021 ITS/08 Web Programming/.NET/Lecture9/WebSite1/" +
6                 "VisitorCounter.dat",
7                 FileMode.OpenOrCreate);
8         int n;
9         try {
10            BinaryReader r = new BinaryReader(s);
11            n = r.ReadInt32();
12        } catch {
13            n = 0;
14        }
15        n++;
16        s.Seek(0, SeekOrigin.Begin);
17        BinaryWriter w = new BinaryWriter(s);
18        w.Write(n);
19        s.Close();
20        return n;
21    }
22 }
```


WebSite2: Automatic creating code file

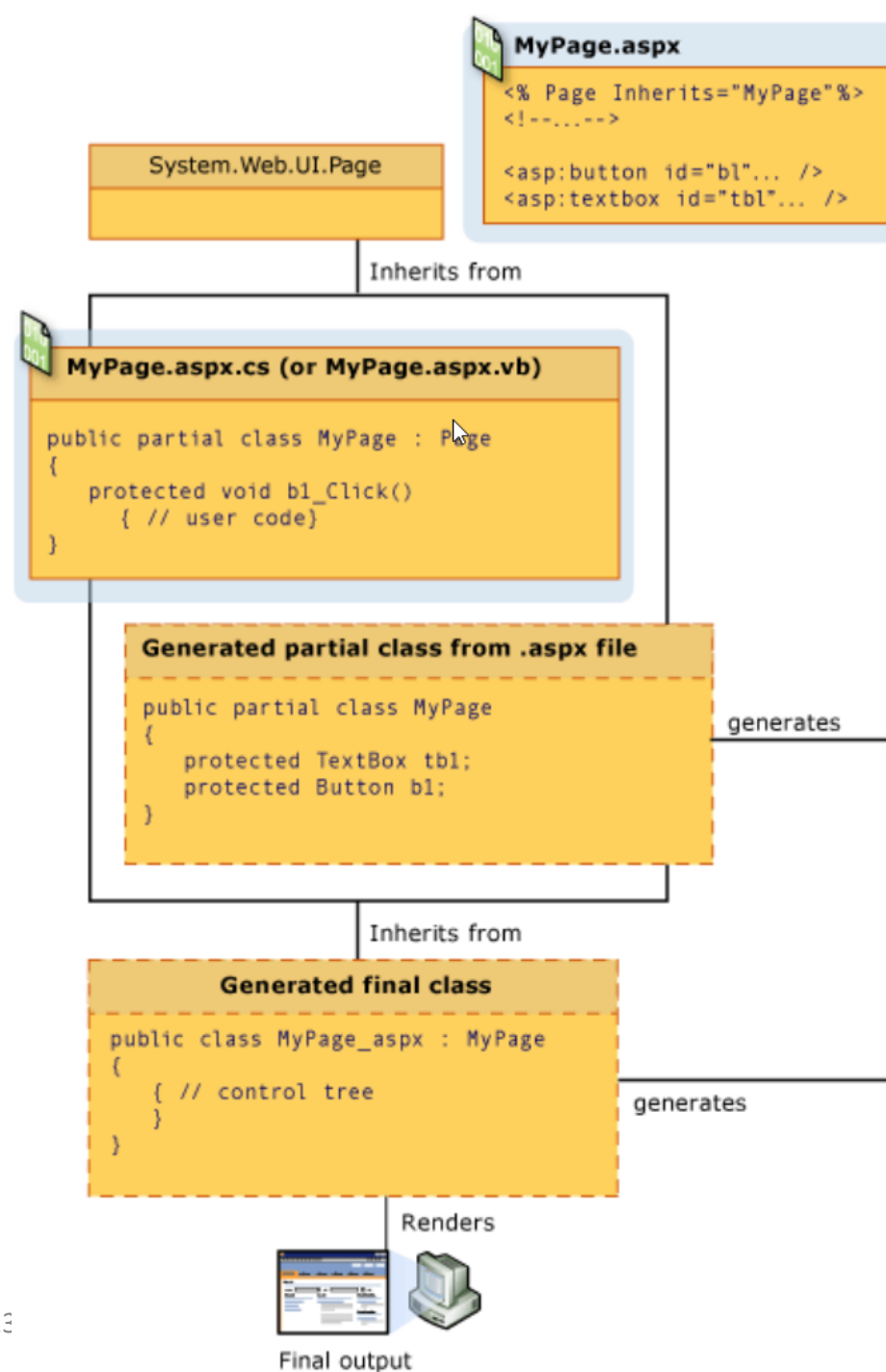
- Whenever VS2019 is creating `.aspx` file, e.g., `MyVisitor2.aspx`, it automatically will create a (source) code file, e.g., `MyVisitor2.aspx.cs`



ASP.NET: Single-file page class



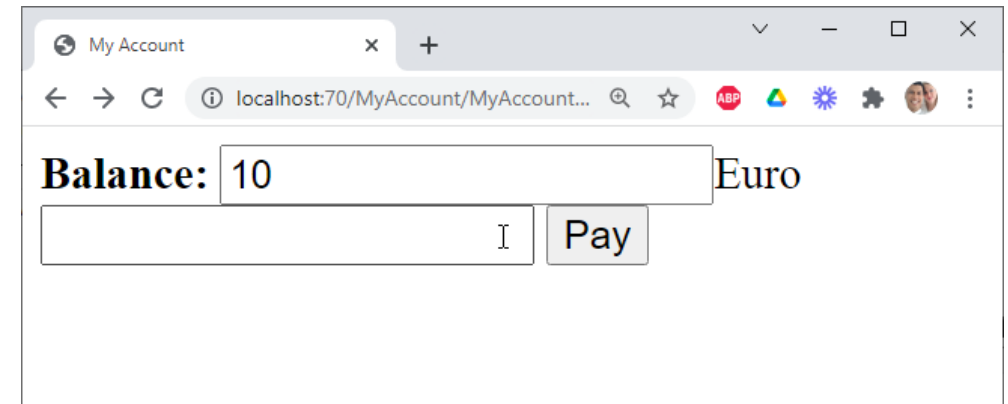
ASP.NET: Code-behind page class



Web form

- Web page usually received input from the user

```
MyAccount.html  x
1  <html>
2  <head>
3    <title>My Account</title>
4  </head>
5  <body>
6    <form action="http://foo.com/cgi-bin/myprog" method="post">
7      <b>Balance:</b>
8      <input type="text" name="total" value="0">Euro<br>
9      <input type="text" name="amount">
10     <input type="submit" name="OK" value="Pay">
11   </form>
12 </body>
13 </html>
```

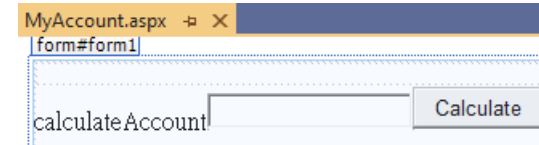


The screenshot shows a web browser window titled "My Account" with the address bar displaying "localhost:70/MyAccount/MyAccount...". The page content includes a form with a label "Balance:" followed by a text input field containing the value "10" and the word "Euro" to its right. Below this is another empty text input field and a "Pay" button.

- Server processes for CGI script `myprog`
 - Read `total` and `amount`
 - Return back the new HTML with the new values of the text `total` and `amount`
- Problem at CGI programming: when a page has to returned back, then the state of text-field has to be set manually → alleviated by ASP.NET & other advanced web programming approaches

MyAccount

- Create new project: MyAccount



```
MyAccount.aspx -> X
1  <%@ Page Language="C#" AutoEventWireup="true"
2      CodeFile="MyAccount.aspx.cs" Inherits="MyAccount" %>
3
4  <!DOCTYPE html>
5  <html xmlns="http://www.w3.org/1999/xhtml">
6  <head runat="server">
7      <title></title>
8  </head>
9  <body>
10 <form id="form1" runat="server">
11     <div>
12     </div>
13     <asp:Label ID="Label1" runat="server" Text="calculateAccount"></asp:Label>
14     <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
15     <asp:Button ID="Button1" runat="server" Text="Calculate" OnClick="calculateAccount" Tooltip="Calculating amount of salary!" />
16     <br />
17     <br />
```

Button1



Properties

Button1 System.Web.UI.WebControls.Button

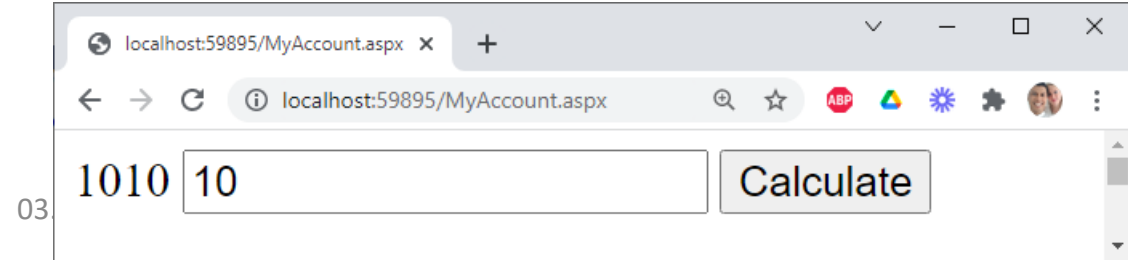
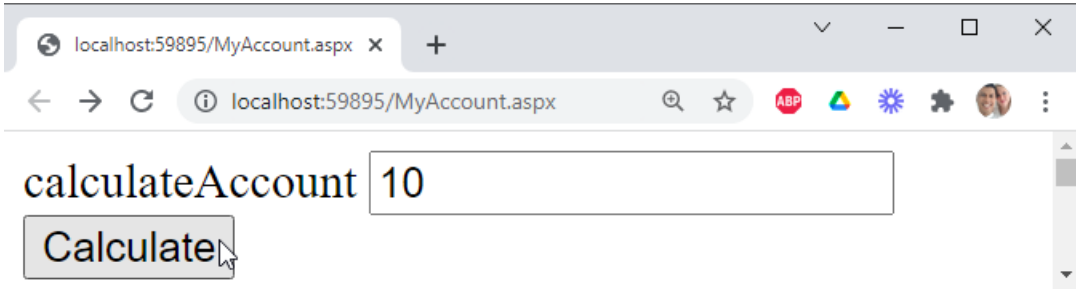
EnableTheming	True
EnableViewState	True
Font	
ForeColor	
Height	
OnClick	
PostBackUrl	
SkinID	
TabIndex	0
Text	Calculate
ToolTip	Calculating amount of salary!

Text

```
1 <%@ Page Language="C#" AutoEventWireup="true"
2   CodeFile="MyAccount.aspx.cs" Inherits="MyAccount" %>
3
4 <!DOCTYPE html>
5 <html xmlns="http://www.w3.org/1999/xhtml">
6 <head runat="server">
7   <title></title>
8 </head>
9 <body>
10  <form id="form1" runat="server">
11    <div>
12    </div>
13    <asp:Label ID="Label1" runat="server" Text="calculateAccount"></asp:Label>
14    <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
15    <asp:Button ID="Button1" runat="server" Text="Calculate" OnClick="calculateAccount" ToolTip="Calculating amount of salary!" />
16    <br />
17    <br />
18  </form>
19 </body>
20 </html>
```

```
MyAccount.aspx.cs - MyAccount.aspx
15_MyAccount.aspx - MyAccount
DoCommand(object sender, Comm

1 using System;
2 using System.Collections;
3 using System.Collections.Generic;
4 using System.Linq;
5 using System.Web;
6 using System.Web.UI;
7 using System.Web.UI.WebControls;
8
9 public partial class MyAccount : System.Web.UI.Page {
10     protected void calculateAccount(object sender, EventArgs e) {
11         int total = 1000;
12         int value = Convert.ToInt32(TextBox1.Text);
13         Label1.Text = (total + value).ToString();
14     }
15 }
```



Button1: Generated page

```
localhost:59895/MyAccount.aspx x view-source:localhost:59895/My/ x +
view-source:localhost:59895/MyAccount.aspx
Line wrap
1
2
3 <!DOCTYPE html>
4 <html xmlns="http://www.w3.org/1999/xhtml">
5 <head><title>
6
7 </title></head>
8 <body>
9   <form method="post" action="./MyAccount.aspx" id="form1">
10 <div class="aspNetHidden">
11 <input type="hidden" name="__EVENTTARGET" id="__EVENTTARGET" value="" />
12 <input type="hidden" name="__EVENTARGUMENT" id="__EVENTARGUMENT" value="" />
13 <input type="hidden" name="__LASTFOCUS" id="__LASTFOCUS" value="" />
14 <input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE" value="kyD02C0L5rzzN5//B5Z1XqfTrV+GWUSecYC/JtnpvBxHMTxkba8mHVrpgeh+K9yailmYZweOFKLnW+fqtcaUM1F0ooHVFdLVQo4hKUnYS00mwJ70PqQoKog4+iKKxcNdZL7/wRbc
15 </div>
16
17 <script type="text/javascript">
18 //
19 var theForm = document.forms['form1'];
20 if (!theForm) {
21   theForm = document.form1;
22 }
23 function __doPostBack(eventTarget, eventArgument) {
24   if (!theForm.onsubmit || (theForm.onsubmit() != false)) {
25     theForm.__EVENTTARGET.value = eventTarget;
26     theForm.__EVENTARGUMENT.value = eventArgument;
27     theForm.submit();
28   }
29 }
30 //]]&gt;
31 &lt;/script&gt;
32
33
34 &lt;div class="aspNetHidden"&gt;
35   &lt;input type="hidden" name="__VIEWSTATEGENERATOR" id="__VIEWSTATEGENERATOR" value="04595DA2" /&gt;
36   &lt;input type="hidden" name="__EVENTVALIDATION" id="__EVENTVALIDATION" value="PhbU58dLfH/WX6s11PpLE1fWkusaA0KfVw8Eew3XXAScVTPB93h7pn4EwMq7sFRstliT4N/F+bEte/BSJmrsF4tezwChosJenRARdMbOydDda2Gy2RPpsxg
37 &lt;/div&gt;
38
39   &lt;div&gt;
40   &lt;/div&gt;
41   &lt;span id="Label1"&gt;1010&lt;/span&gt;
42   &lt;input name="TextBox1" type="text" value="10" id="TextBox1" /&gt;
43   &lt;input type="submit" name="Button1" value="Calculate" id="Button1" title="Calculating amount of salary!" /&gt;
44   &lt;br /&gt;
45   &lt;br /&gt;</pre></div>
```

ASP.NET: General notation

```
<asp:ClassName propertyName="value" ... Runat="server" />
```

- **E.g.**, `<asp:Label ID="total" Text="Hello" ForeColor="Blue" Runat="server" />`
- **In web server, this tag will be generated into a class, where all of the web control classes under namespace `System.Web.UI`**

```
public class Label: WebControl {  
    public virtual string ID { get { ... } set { ... } }  
    public virtual string Text { get { ... } set { ... } }  
    public virtual Color ForeColor { get { ... } set { ... } }  
    ...  
}
```

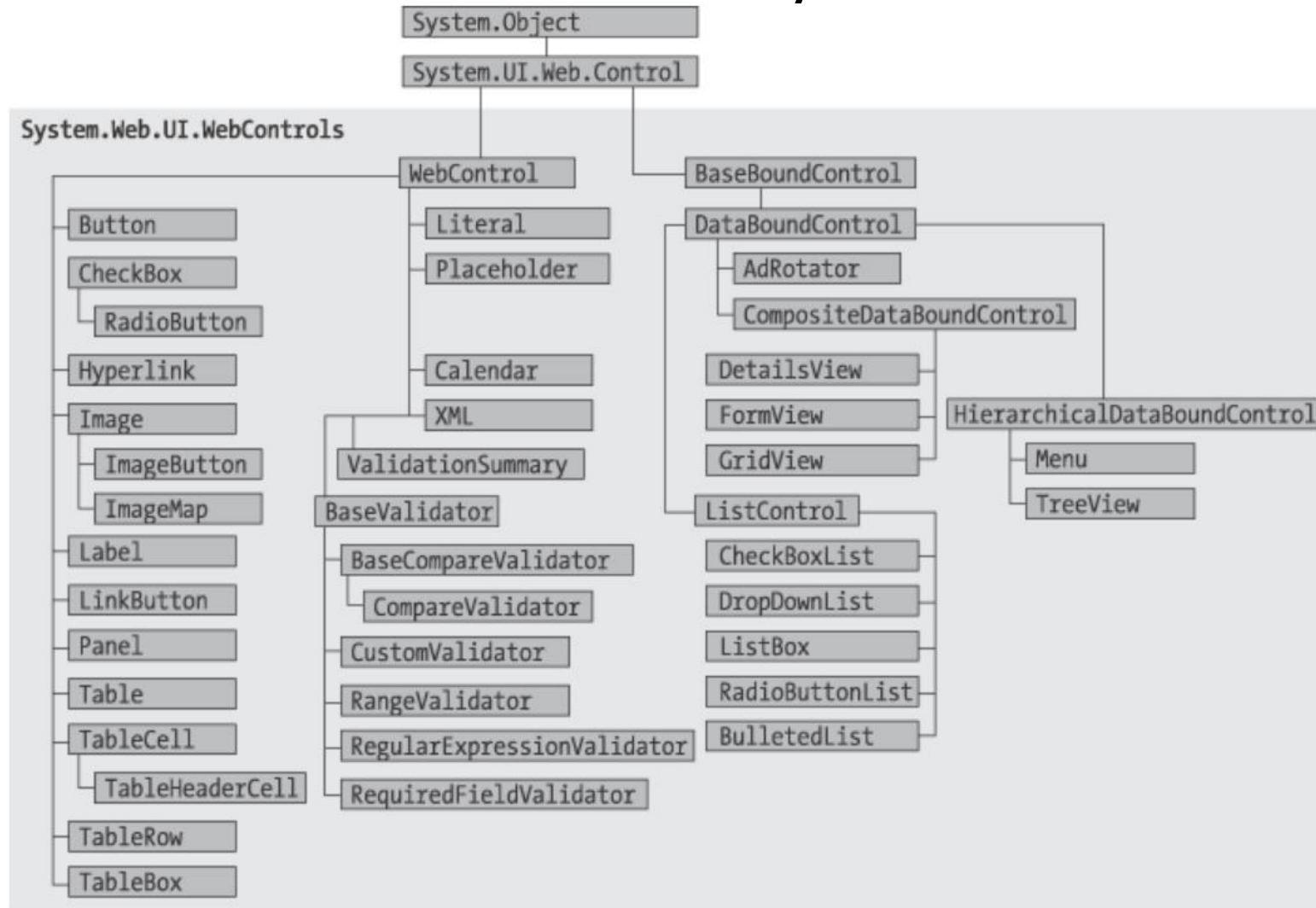
- **Alternatively,**

```
<asp:Label ID="total" ForeColor="Blue" Runat="server" />  
    Hello  
</asp:Label>
```


ASP.NET: Web form advantage

- “Traditional” ASP and other “old” program: web server interprets script of a page file and run it one by one, so that web server has been burdened with this approach
- ASP.NET & other advanced approaches: the web server separated with the framework → web server tasks will be distributed into the framework which proceed the translation a code to be an object
 - A web page → an object with its properties and methods
 - GUI element of an object → can be accessed by its methods

WebControl: The hierarchy



Event handling

- An event is an action or occurrence such as a mouse click, a key press, mouse movements, or any system-generated notification. A process communicates through events. For example, interrupts are system-generated events. When events occur, the application should be able to respond to it and manage it.
- Events in ASP.NET raised at the client machine, and handled at the server machine. For example, a user clicks a button displayed in the browser. A Click event is raised. The browser handles this client-side event by posting it to the server.
- The server has a subroutine describing what to do when the event is raised; it is called the event-handler. Therefore, when the event message is transmitted to the server, it checks whether the Click event has an associated event handler. If it has, the event handler is executed.

Event Arguments

- ASP.NET event handlers generally take two parameters and return void. The first parameter represents the object raising the event and the second parameter is event argument.
- The general syntax of an event is:

```
private void EventName (object sender, EventArgs e);
```

Application and session events

- The most important application events are:
 - Application_Start - It is raised when the application/website is started.
 - Application_End - It is raised when the application/website is stopped.
- Similarly, the most used Session events are:
 - Session_Start - It is raised when a user first requests a page from the application.
 - Session_End - It is raised when the session ends.

Page and control events

- Common page and control events are:
 - DataBinding - It is raised when a control binds to a data source.
 - Disposed - It is raised when the page or the control is released.
 - Error - It is a page event, occurs when an unhandled exception is thrown.
 - Init - It is raised when the page or the control is initialized.
 - Load - It is raised when the page or a control is loaded.
 - PreRender - It is raised when the page or the control is to be rendered.
 - Unload - It is raised when the page or control is unloaded from memory.

Event handling using controls

- The ASP tag for a button control:

```
<asp:Button ID="Button1" runat="server" Text="Calculate"  
    OnClick="calculateAccount" Tooltip="Calculating amount of salary!" />
```

- The event handler for the Click event:

```
protected void calculateAccount(object sender, EventArgs e) {  
    int total = 1000;  
    int value = Convert.ToInt32(TextBox1.Text);  
    Label1.Text = (total + value).ToString();  
}
```

- An event can also be coded without Handles clause. Then, the handler must be named according to the appropriate event attribute of the control.

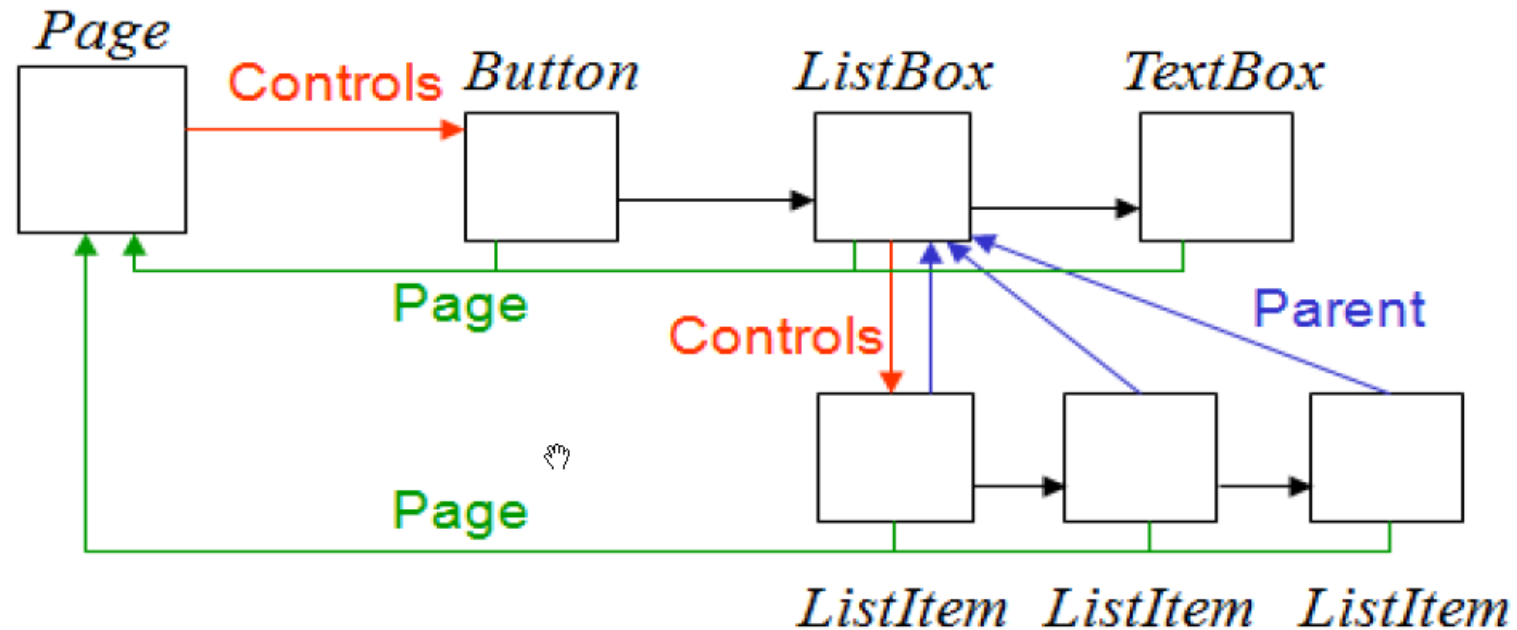
Common control events

Event	Attribute	Controls
Click	OnClick	Button, image button, link button, image map
Command	OnCommand	Button, image button, link button
TextChanged	OnTextChanged	Text box
SelectedIndexChanged	OnSelectedIndexChanged	Drop-down list, list box, radio button list, check box list.
CheckedChanged	OnCheckedChanged	Check box, radio button

Default events

Control	Default Event
AdRotator	AdCreated
BulletedList	Click
Button	Click
Calender	SelectionChanged
CheckBox	CheckedChanged
CheckBoxList	SelectedIndexChanged
DataGrid	SelectedIndexChanged
DataList	SelectedIndexChanged
DropDownList	SelectedIndexChanged
HyperLink	Click
ImageButton	Click
ImageMap	Click
LinkButton	Click
ListBox	SelectedIndexChanged
Menu	MenuItemClick
RadioButton	CheckedChanged
RadioButtonList	SelectedIndexChanged

Web control: Classes example



- A *Page* has *Button*, *ListBox*, *TextBox* controls
- *ListBox* has *ListItem* control → *ListItem* has *ListBox* as its parent

Web form & control: Example

```
MyAccount.aspx - x
1  <%@ Page Language="C#" AutoEventWireup="true"
2    CodeFile="MyAccount.aspx.cs" Inherits="MyAccount" %>
3
4  <!DOCTYPE html>
5  <html xmlns="http://www.w3.org/1999/xhtml">
6  <head runat="server">
7    <title></title>
8  </head>
9  <body>
10 <form id="form1" runat="server">
11   <div>
12   </div>
13   <asp:Label ID="Label1" runat="server"
14     Text="calculateAccount"></asp:Label>
15   <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
16   <asp:Button ID="Button1" runat="server" Text="Calculate"
17     onClick="calculateAccount" Tooltip="Calculating amount of salary!" />
18   <br />
19   <br />
20   <asp:Label ID="myLabel" runat="server" Text="100"></asp:Label>
21   <asp:Button ID="plus10" runat="server" Text="+ 10%"
22     CommandName="add" CommandArgument="0.1" OnCommand="DoCommand"/>
23   <asp:Button ID="min5" runat="server" Text="- 5%"
24     CommandName="sub" CommandArgument="0.05" OnCommand="DoCommand"/>
25   <br />
26   <br />
27   <asp:TextBox ID="normalTextBox" Text="Sample" runat="server" />
28   <asp:TextBox ID="passwordTextBox" TextMode="Password"
29     MaxLength="10" runat="server"></asp:TextBox>
30   <asp:TextBox ID="multilineTextBox" TextMode="MultiLine"
31     Rows="3" Columns="15" Wrap="true"
32     runat="server">Durian Mango Mangosteen Pineapple</asp:TextBox>
33   <br />
34   <br />
35   <asp:CheckBox ID="apples" Text="Apples" runat="server" />
36   <br />
37   <asp:CheckBox ID="pears" Text="Pears" runat="server" />
38   <br />
39   <asp:CheckBox ID="bananas" Text="Bananas" runat="server" />
40   <br />
41   <br />
42   <asp:Button ID="btnBuy" runat="server" Text="Buy"
43     onClick="DoClick"/>
44   <br />
45   <asp:Label ID="lblBuy" runat="server" Text="Label"></asp:Label>
46   <br />
47   <br />
48   <asp:Label ID="Label2" runat="server"
49     Text="Select method of payment"></asp:Label>
50   <br />
51   <asp:RadioButton ID="rbCash" runat="server" Text="cash"
52     GroupName="payment" OnCheckedChanged="RadioChanged"
53     AutoPostBack="true"/>
54   <br />
55   <asp:RadioButton ID="rbCheque" runat="server" Text="cheque"
56     GroupName="payment" OnCheckedChanged="RadioChanged"
57     AutoPostBack="true"/>
58   <br />
59   <asp:RadioButton ID="rbCC" runat="server" Text="credit card"
60     GroupName="payment" OnCheckedChanged="RadioChanged"
61     AutoPostBack="true"/>
```

Web form & control: Example (continued)

```
62 <br />
63 <br />
64 <asp:Label ID="lblPayment" runat="server" Text="Label"></asp:Label>
65 <br />
66 <br />
67 <asp:ListBox ID="lbCarCode" runat="server"           86 <br />
68   Rows="3" SelectionMode="Multiple">           87 <br />
69   <asp:ListItem Text="United States" Value="USA"   88 <asp:ListBox ID="lbCarCode2" runat="server" AutoPostBack="true"
70     Runat="server" />           89   OnSelectedIndexChanged="lbCarCode2_SelectedIndexChanged"></asp:ListBox>
71   <asp:ListItem Text="Great Britain" Value="GB"   90 <br />
72     Runat="server" />           91 <asp:Button ID="btnCarCode2" runat="server" Text="Fill"
73   <asp:ListItem Text="Germany" Value="D"         92   OnClick="btnCarCode2_Click" />
74     Runat="server" />           93 <br />
75   <asp:ListItem Text="France" Value="F"         94 <br />
76     Runat="server" />           95 <asp:Label ID="lblCarCode2" runat="server" Text="Label"></asp:Label>
77   <asp:ListItem Text="Italy" Value="I"           96 <br />
78     Runat="server" />           97 </form>
79 </asp:ListBox>           98 </body>
80 <br />           99 </html>
81 <br />
82 <asp:Button ID="btnCarCode" runat="server" Text="Show"
83   OnClick="CarCodeButtonClick"/>
84 <br />
85 <asp:Label ID="lblCarCode" runat="server" Text="Label"></asp:Label>
```

Web form & control: Example (continued)

localhost:59895/MyAccount.aspx x +

localhost:59895/MyAccount.aspx ☆ ABP

calculateAccount Calculate

100 + 10% - 5%

Sample

Durian Mango
Mangosteen
Pineapple

Apples
 Pears
 Bananas

Buy

Label

Select method of payment

cash
 cheque
 credit card

Label

United States
Great Britain
Germany

Show

Label

Fill

Label

Web form & control: Example (continued)

```
MyAccount.aspx.cs  x
16_MyAccount.aspx  MyAccount

1  using System;
2  using System.Collections;
3  using System.Collections.Generic;
4  using System.Linq;
5  using System.Web;
6  using System.Web.UI;
7  using System.Web.UI.WebControls;
8
9  public partial class MyAccount : System.Web.UI.Page {
10     protected void calculateAccount(object sender, EventArgs e) {
11         int total = 1000;
12         int value = Convert.ToInt32(TextBox1.Text);
13         Label1.Text = (total + value).ToString();
14     }
15     public void DoCommand(object sender, CommandEventArgs e) {
16         double total = Convert.ToDouble(myLabel.Text);
17         if (e.CommandName == "add") {
18             total += total * Convert.ToDouble(e.CommandArgument);
19         } else if (e.CommandName == "sub") {
20             total -= total * Convert.ToDouble(e.CommandArgument);
21         }
22         myLabel.Text = total.ToString("F2");
23     }
24     protected void DoClick(object sender, EventArgs e) {
25         lblBuy.Text = "You bought: ";
26         if (apples.Checked) {
27             lblBuy.Text += "Apples ";
28         }
29         if (pears.Checked) {
30             lblBuy.Text += "Pears ";
31         }
32         if (bananas.Checked) {
33             lblBuy.Text += "Bananas ";
34         }
35     }
36     protected void RadioChanged(object sender, EventArgs e) {
37         lblPayment.Text = "Method of payment: ";
38         if (rbCash.Checked) {
39             lblPayment.Text += rbCash.Text;
40         }
41         if (rbCheque.Checked) {
42             lblPayment.Text += rbCheque.Text;
43         }
44         if (rbCC.Checked) {
45             lblPayment.Text += rbCC.Text;
46         }
47     }
}
```

Web form & control: Example

```
48 protected void CarCodeButton_Click(object sender, EventArgs e) {
49     lblCarCode.Text = "The selected country has the international car code ";
50     if (lbCarCode.SelectedItem != null) {
51         lblCarCode.Text += lbCarCode.SelectedItem.Value;
52     }
53 }
54
55 protected void btnCarCode2_Click(object sender, EventArgs e) {
56     SortedList data = new SortedList();
57     data["United States"] = "USA";
58     data["Great Britain"] = "GB";
59     data["Germany"] = "G";
60     data["France"] = "F";
61     data["Italy"] = "I";
62     lbCarCode2.DataSource = data;
63     lbCarCode2.DataTextField = "Key";
64     lbCarCode2.DataValueField = "Value";
65     lbCarCode2.DataBind();
66 }
67
68 protected void lbCarCode2_SelectedIndexChanged(object sender, EventArgs e) {
69     lblCarCode2.Text = "The selected country has the " +
70         "international car code ";
71     if (lbCarCode2.SelectedItem != null) {
72         lblCarCode2.Text += lbCarCode2.SelectedItem.Value;
73     }
74 }
75 }
76
```

The screenshot shows a web browser window at localhost:59895/MyAccount.aspx. The page contains several form elements:

- A text input field containing "1250" and a "Calculate" button.
- Two buttons labeled "+ 10%" and "- 5%".
- A list box containing "Durian Mango", "Mangosteen", and "Pineapple".
- Two text input fields, one containing "Sample".
- Three checked checkboxes: "Apples", "Pears", and "Bananas".
- A "Buy" button.
- Text: "You bought: Appples Bananas".
- Section: "Select method of payment" with radio buttons for "cash", "cheque" (selected), and "credit card".
- Text: "Method of payment: cheque".
- A dropdown menu with "United States", "Great Britain" (selected), and "Germany".
- A "Show" button.
- Text: "The selected country has the international car code GB".
- Another dropdown menu with "France", "Germany" (selected), "Great Britain", and "Italy".
- A "Fill" button.
- Text: "The selected country has the international car code G".