

Lesson 2-12: Work with HTML Forms

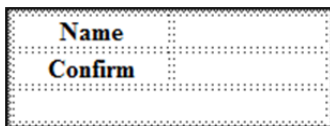
HTML forms are the only mechanism available in HTML that is able to send information to the web server. All of the buttons and text boxes you've added in previous lessons have been *form controls*.

HTML forms are relatively simple, but they are the backbone of ASP.NET.

- 1 Open *HTMLTest* from your sample files folder.
- 2 Open *formstest.htm* in *Design* view.

An *.htm* file is a pure HTML file that doesn't contain any of ASP.NET's special features. Browsers will recognize HTML files with both the *.htm* and *.html* file extensions but *.htm* is the file extension favored by Visual Studio.

You should recognize the design as an HTML table with some text in the cells. This is going to be your form.



- 3 Add an HTML Text input control.

1. Add an *Input (Text)* control from the *HTML* category of the *Toolbox* to the cell to the right of *Name*.

The *Input (Text)* control is the HTML equivalent of the ASP.NET *TextBox* control, which you'll learn about in: *Lesson 4-4: Use text boxes*.



2. Set the *Name* property of the new control to: **Name**

The *Name* property is used to identify the data that will be sent by the form (see sidebar).



- 4 Add an HTML Checkbox input control.

1. In the same way as you did in the last step, add an *Input (Checkbox)* control in the cell next to *Confirm*.

The *Input (Checkbox)* control is the HTML equivalent of the ASP.NET *CheckBox* control, which you'll learn about in: *Lesson 4-5: Use check boxes*.



2. Set the *Name* property of the new checkbox to: **Confirm**



- 5 Add an HTML Submit input control.

note

The Name property

You have to set the *Name* property in this lesson because you're using a pure HTML form.

When you start using ASP.NET's controls later on you won't need to worry about the *Name* property anymore because ASP.NET will handle it automatically.

note

Unix and Windows Web servers

Pure HTML code such as you see in this lesson can be run on any web server. That's not true of ASP.NET code, however.

ASP.NET web sites must be hosted on a server running the Windows operating system and Microsoft's IIS web server software.

Many web hosting services on the Internet use the open-source Linux operating system and Apache web server software instead. There's a big cost saving associated with this approach as the web hosting service doesn't have to purchase any software (Linux and Apache are both freeware).

While Apache doesn't support ASP.NET it does support a similar (also open-source) set of technologies based upon the PHP programming language.

PHP is an open source alternative to ASP.NET which is favored by hobbyists.

Many web bulletin boards can be found with passionate arguments about whether PHP or ASP.NET is "best". There's no real answer to this question but many agree that PHP is more suitable for smaller projects while ASP.NET makes it easier to work with large and complex projects.

Add an *Input (Submit)* control from the *HTML* category of the *Toolbox* to the bottom cell of the table.

The *Input (Submit)* HTML control is used to send the contents of the form to the server. It's the equivalent of the ASP.NET *Button* control, which you'll learn more about in: *Lesson 4-2: Use button controls*.

6 Set the *action* of the form.

1. Switch to *Source* view.
2. You'll see that there are no *asp* elements on this page at all. This is a pure HTML page that could be served by any web server to any web browser (see sidebar).
3. Set the *action* property of the *form* tag to: **formsubmit.aspx**

You can do this by adding the property to the tag using the code **action="formsubmit.aspx"** or by clicking on the *form* tag and using the *Properties* window to set the *Action* property.

```
<form id="form1" action="formsubmit.aspx">
```

The *action* property tells the form where to send the data that the user has entered when the submit button is clicked.

formsubmit.aspx is an ASP.NET page that will process the data and display it.

When using *aspx* pages, you'll never need to manually set the *action* property. You'll learn more about how ASP.NET sends and receives data in: *Lesson 3-11: Send data between pages*.

7 Submit the form.

1. View *formstest.htm* in your browser.
2. Fill in the form and click *submit*.

You are redirected to *formsubmit.aspx*, which receives the data you entered and displays it.

```
Your name is: Simon Smart
You ticked confirm.
```

Almost everything ASP.NET does revolves around this system of HTML forms sending data to the web server. Visual Studio generates most of the HTML code that makes this work automatically, but with the knowledge you've gained from this lesson you should be able to understand what it is doing behind the scenes.