

FTP (File Transfer Protocol)

One of the most common methods for sending files between two computers.

Character Entities

Some characters have a special meaning in HTML, like the less than sign (<) that defines the start of an HTML tag. If we want the browser to actually display these characters we must insert character entities in the HTML source.

A character entity has three parts: an ampersand (&), an entity name or a # and an entity number, and finally a semicolon (;).

To display a less than sign in an HTML document we must write: **<** or **<**;

The advantage of using a name instead of a number is that a name is easier to remember. The disadvantage is that not all browsers support the newest entity names, while the support for entity numbers is very good in almost all browsers.

Note that the entities are case sensitive.

This example lets you experiment with character entities: [Character Entities](#) IE only

Non-breaking Space

The most common character entity in HTML is the non-breaking space.

Normally HTML will truncate spaces in your text. If you write 10 spaces in your text HTML will remove 9 of them. To add spaces to your text, use the ` ` character entity.

The Most Common Character Entities:

Result	Description	Entity Name	Entity Number
	non-breaking space	<code>&nbsp;</code>	<code>&#160;</code>
<	less than	<code>&lt;</code>	<code>&#60;</code>
>	greater than	<code>&gt;</code>	<code>&#62;</code>
&	ampersand	<code>&amp;</code>	<code>&#38;</code>

"	quotation mark	"	"
'	apostrophe	' (does not work in IE)	'

Some Other Commonly Used Character Entities:

Result	Description	Entity Name	Entity Number
¢	cent	¢	¢
£	pound	£	£
¥	yen	¥	¥
§	section	§	§
©	copyright	©	©
®	registered trademark	®	®
×	multiplication	×	×
÷	division	÷	÷

DOC Types

The XHTML standard defines three Document Type Definitions.

The most common is the XHTML Transitional.

<!DOCTYPE> Is Mandatory

An XHTML document consists of three main parts:

- the DOCTYPE
- the Head
- the Body

The basic document structure is:

```
<!DOCTYPE ...>
<html>
<head>
<title>... </title>
</head>
<body> ... </body>
</html>
```

The DOCTYPE declaration should always be the first line in an XHTML document.

An XHTML Example

This is a simple (minimal) XHTML document:

```
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
<head>
<title>simple document</title>
</head>
<body>
<p>a simple paragraph</p>
</body>
</html>
```

The DOCTYPE declaration defines the document type:

```
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

The rest of the document looks like HTML:

```
<html>
<head>
<title>simple document</title>
</head>
<body>
<p>a simple paragraph</p>
</body>
</html>
```

The 3 Document Type Definitions

- DTD specifies the syntax of a web page in SGML.
- DTD is used by SGML applications, such as HTML, to specify rules that apply to the markup of documents of a particular type, including a set of element and entity declarations.
- XHTML is specified in an SGML document type definition or 'DTD'.
- An XHTML DTD describes in precise, computer-readable language, the allowed syntax and grammar of XHTML markup.

There are currently 3 XHTML document types:

- STRICT
- TRANSITIONAL
- FRAMESET

XHTML 1.0 specifies three XML document types that correspond to three DTDs: Strict, Transitional, and Frameset.

XHTML 1.0 Strict

```
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

Use this when you want really clean markup, free of presentational clutter. Use this together with Cascading Style Sheets.

XHTML 1.0 Transitional

```
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

Use this when you need to take advantage of HTML's presentational features and when you want to support browsers that don't understand Cascading Style Sheets.

XHTML 1.0 Frameset

```
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
```

Use this when you want to use HTML Frames to partition the browser window into two or more frames.

HTML and XHTML Links

HTML uses a hyperlink to link to another document on the Web.

Examples

[Create hyperlinks](#)

This example demonstrates how to create links in an HTML document.

[An image as a link](#)

This example demonstrates how to use an image as a link.

(You can find more examples at the bottom of this page)

The Anchor Tag and the Href Attribute

HTML uses the <a> (anchor) tag to create a link to another document.

An anchor can point to any resource on the Web: an HTML page, an image, a sound file, a movie, etc.

The syntax of creating an anchor:

```
<a href="url">Text to be displayed</a>
```

The <a> tag is used to create an anchor to link from, the href attribute is used to address the document to link to, and the words between the open and close of the anchor tag will be displayed as a hyperlink.

This anchor defines a link to W3Schools:

```
<a href="http://www.w3schools.com/">Visit W3Schools!</a>
```

The line above will look like this in a browser:

[Visit W3Schools!](http://www.w3schools.com/)

The Target Attribute

With the target attribute, you can define **where** the linked document will be opened.

The line below will open the document in a new browser window:

```
<a href="http://www.w3schools.com/"  
target="_blank">Visit W3Schools!</a>
```

The Anchor Tag and the Name Attribute

The name attribute is used to create a named anchor. When using named anchors we can create links that can jump directly into a specific section on a page, instead of letting the user scroll around to find what he/she is looking for.

Below is the syntax of a named anchor:

```
<a name="label">Text to be displayed</a>
```

The name attribute is used to create a named anchor. The name of the anchor can be any text you care to use.

The line below defines a named anchor:

```
<a name="tips">Useful Tips Section</a>
```

You should notice that a named anchor is not displayed in a special way.

To link directly to the "tips" section, add a # sign and the name of the anchor to the end of a URL, like this:

```
<a href="http://www.w3schools.com/html_links.asp#tips">  
Jump to the Useful Tips Section</a>
```

A hyperlink to the Useful Tips Section from WITHIN the file "html_links.asp" will look like this:

```
<a href="#tips">Jump to the Useful Tips Section</a>
```

Basic Notes - Useful Tips

Always add a trailing slash to subfolder references. If you link like this: href="http://www.w3schools.com/html", you will generate two HTTP requests to the server, because the server will add a slash to the address and create a new request like this: href="http://www.w3schools.com/html/"

Named anchors are often used to create "table of contents" at the beginning of a large document. Each chapter within the document is given a named anchor, and links to each of these anchors are put at the top of the document.

If a browser cannot find a named anchor that has been specified, it goes to the top of the document. No error occurs.

Link Tags

Tag	Description
<a>	Defines an anchor

More Examples

[Open a link in a new browser window](#)

This example demonstrates how to link to another page by opening a new window, so that the visitor does not have to leave your Web site.

```
<a href="lastpage.htm" target="_blank">Last Page</a>
```

[Link to a location on the same page](#)

This example demonstrates how to use a link to jump to another part of a document.

```
<p>  
    <a href="#C4">See also Chapter 4.</a>  
</p>
```

```
<h2>Chapter 1</h2>  
<p>This chapter explains ba bla bla</p>
```

```
<h2>Chapter 2</h2>  
<p>This chapter explains ba bla bla</p>
```

```
<h2>Chapter 3</h2>  
<p>This chapter explains ba bla bla</p>
```

```
<h2><a name="C4">Chapter 4</a></h2>  
<p>This chapter explains ba bla bla</p>
```

```
<h2>Chapter 5</h2>  
<p>This chapter explains ba bla bla</p>
```

[Create a mailto link](#)

This example demonstrates how to link to a mail message (will only work if you have mail installed).

```
<a href="mailto:someone@microsoft.com?subject=Hello%20again">Send  
Mail</a>
```

[Create a mailto link 2](#)

This example demonstrates a more complicated mailto link.

```
<a  
href="mailto:someone@microsoft.com?cc=someoneelse@microsoft.com&bcc=andsome  
oneelse2@microsoft.com&subject=Summer%20Party&body=You%20are%20invited%2  
0to%20a%20big%20summer%20party!">Send mail!</a>
```