

# HTML(5) Style Guide and Coding Conventions

[« Previous](#)

[Next Chapter »](#)

---

## HTML Coding Conventions

Web developers are often uncertain about the coding style and syntax to use in HTML.

Between 2000 and 2010, many web developers converted from HTML to XHTML.

With XHTML, developers were forced to write valid and "well-formed" code.

HTML5 is a bit more sloppy when it comes to code validation.

With HTML5, you must create your own **Best Practice, Style Guide and Coding Conventions**.

---

## Be Smart and Future Proof

A consequent use of style, makes it easier for others to understand and use your HTML.

In the future, programs like XML readers, may want to read your HTML.

Using a well-formed "close to XHTML" syntax, can be smart.

Always keep your style smart, tidy, clean, and well-formed.

---

## Use Correct Document Type

Always declare the document type as the first line in your document:

```
<!DOCTYPE html>
```

If you want consistency with lower case tags, you can use:

```
<!doctype html>
```

---

## Use Lower Case Element Names

HTML5 allows mixing uppercase and lowercase letters in element names.

We recommend using lowercase element names:

- Mixing uppercase and lowercase names is bad
- Developers are used to using lowercase names (as in XHTML)
- Lowercase look cleaner
- Lowercase are easier to write

### Bad:

```
<SECTION>  
  <p>This is a paragraph.</p>  
</SECTION>
```

### Very Bad:

```
<Section>
  <p>This is a paragraph.</p>
</SECTION>
```

## Good:

```
<section>
  <p>This is a paragraph.</p>
</section>
```

---

## Close All HTML Elements

In HTML5, you don't have to close all elements (for example the `<p>` element).

We recommend closing all HTML elements:

Looking bad:

```
<section>
  <p>This is a paragraph.
  <p>This is a paragraph.
</section>
```

Looking good:

```
<section>
  <p>This is a paragraph.</p>
  <p>This is a paragraph.</p>
</section>
```

---

## Close Empty HTML Elements

In HTML5, it is optional to close empty elements.

This is allowed:

```
<meta charset="utf-8">
```

This is also allowed:

```
<meta charset="utf-8" />
```

The slash (/) is required in XHTML and XML.

If you expect XML software to access your page, it might be a good idea to keep it.

---

## Use Lower Case Attribute Names

HTML5 allows mixing uppercase and lowercase letters in attribute names.

We recommend using lowercase attribute names:

- Mixing uppercase and lowercase names is bad
- Developers are used to using lowercase names (as in XHTML)
- Lowercase look cleaner
- Lowercase are easier to write

Looking bad:

```
<div CLASS="menu">
```

Looking good:

```
<div class="menu">
```

---

## Quote Attribute Values

HTML5 allows attribute values without quotes.

We recommend quoting attribute values:

- You have to use quotes if the value contains spaces
- Mixing styles is never good

- Quoted values are easier to read

This will not work, because the value contains spaces:

```
<table class=table striped>
```

This will work:

```
<table class="table striped">
```

---

## Image Attributes

Always use the **alt** attribute with images. It is important when the image cannot be viewed.

```

```

Always define image size. It reduces flickering because the browser can reserve space for images before they are loaded.

```

```

---

## Spaces and Equal Signs

Spaces around equal signs is legal:

```
<link rel = "stylesheet" href = "styles.css">
```

But space-less is easier to read, and groups entities better together:

```
<link rel="stylesheet" href="styles.css">
```

---

## Avoid Long Code Lines

When using an HTML editor, it is inconvenient to scroll right and left to read the HTML code.

Try to avoid code lines longer than 80 characters.

---

## Blank Lines and Indentation

Do not add blank lines without a reason.

For readability, add blank lines to separate large or logical code blocks.

For readability, add 2 spaces of indentation. Do not use TAB.

Do not use unnecessary blank lines and indentation. It is not necessary to use blank lines between short and related items. It is not necessary to indent every element:

### Unnecessary:

```
<body>

  <h1>Famous Cities</h1>

  <h2>Tokyo</h2>

  <p>
    Tokyo is the capital of Japan, the center of the Greater Tokyo
    Area,
    and the most populous metropolitan area in the world.
    It is the seat of the Japanese government and the Imperial Palace,
    and the home of the Japanese Imperial Family.
  </p>

</body>
```

### Better:

```
<body>

<h1>Famous Cities</h1>

<h2>Tokyo</h2>
<p>Tokyo is the capital of Japan, the center of the Greater Tokyo
Area,
and the most populous metropolitan area in the world.
It is the seat of the Japanese government and the Imperial Palace,
and the home of the Japanese Imperial Family.</p>

</body>
```

## Table Example:

```
<table>
  <tr>
    <th>Name</th>
    <th>Description</th>
  </tr>
  <tr>
    <td>A</td>
    <td>Description of A</td>
  </tr>
  <tr>
    <td>B</td>
    <td>Description of B</td>
  </tr>
</table>
```

## List Example:

```
<ol>
  <li>London</li>
  <li>Paris</li>
  <li>Tokyo</li>
</ol>
```

---

# Omitting <html> and <body>?

In the HTML5 standard, the <html> tag and the <body> tag can be omitted.

The following code will validate as HTML5:

## Example

```
<!DOCTYPE html>
<head>
  <title>Page Title</title>
</head>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>
```

Try it Yourself »

**We do not recommend omitting the <html> and <body> tags.**

The <html> element is the document root. It is the recommended place for specifying the page language:

```
<!DOCTYPE html>
<html lang="en-US">
```

Declaring a language is important for accessibility applications (screen readers) and search engines.

Omitting <html> or <body> can crash DOM and XML software.

Omitting <body> can produce errors in older browsers (IE9).

---

## Omitting <head>?

In the HTML5 standard, the <head> tag can also be omitted.



By default, browsers will add all elements before `<body>`, to a default `<head>` element.

You can reduce the complexity of HTML, by omitting the `<head>` tag:

## Example

```
<!DOCTYPE html>
<html>
<title>Page Title</title>

<body>
  <h1>This is a heading</h1>
  <p>This is a paragraph.</p>
</body>

</html>
```

Try it Yourself »

Omitting tags is unfamiliar to web developers. It needs time to be established as a guideline.

---

## Meta Data

The `<title>` element is required in HTML5. Make the title as meaningful as possible:

```
<title>HTML5 Syntax and Coding Style</title>
```

To ensure proper interpretation, and correct search engine indexing, both the language and the character encoding should be defined as early as possible in a document:

```
<!DOCTYPE html>
<html lang="en-US">
<head>
  <meta charset="UTF-8">
```

```
<title>HTML5 Syntax and Coding Style</title>
</head>
```

---

## HTML Comments

Short comments should be written on one line, with a space after `<!--` and a space before `-->`:

```
<!-- This is a comment -->
```

Long comments, spanning many lines, should be written with `<!--` and `-->` on separate lines:

```
<!--
  This is a long comment example. This is a long comment example.
  This is a long comment example.
  This is a long comment example. This is a long comment example.
  This is a long comment example.
-->
```

Long comments are easier to observe, if they are indented 2 spaces.

---

## Style Sheets

Use simple syntax for linking style sheets (the type attribute is not necessary):

```
<link rel="stylesheet" href="styles.css">
```

Short rules can be written compressed, on one line, like this:

```
p.into {font-family: Verdana; font-size: 16em;}
```

Long rules should be written over multiple lines:

```
body {
  background-color: lightgrey;
  font-family: "Arial Black", Helvetica, sans-serif;
```

```
font-size: 16em;  
color: black;  
}
```

- Place the opening bracket on the same line as the selector.
- Use one space before the opening bracket.
- Use 2 spaces of indentation.
- Use colon plus one space between each property and its value.
- Use space after each comma or semicolon.
- Use semicolon after each property-value pair, including the last.
- Only use quotes around values if the value contains spaces.
- Place the closing bracket on a new line, without leading spaces.
- Avoid lines over 80 characters.

Adding a space after a comma, or a semicolon, is a general rule in all types of writing.

---

## Loading JavaScript in HTML

Use simple syntax for loading external scripts (the type attribute is not necessary):

```
<script src="myscript.js">
```

---

## Accessing HTML Elements with JavaScript

A consequence of using "untidy" HTML styles, might result in JavaScript errors.

These two JavaScript statements will produce different results:

### Example

```
var obj = getElementById("Demo")
```

```
var obj = getElementById("demo")
```

Try it Yourself »

If possible, use the same naming convention (as JavaScript) in HTML.

[Visit the JavaScript Style Guide.](#)

---

## Use Lower Case File Names

Most web servers (Apache, Unix) are case sensitive about file names:

london.jpg cannot be accessed as London.jpg.

Other web servers (Microsoft, IIS) are not case sensitive:

london.jpg can be accessed as London.jpg or london.jpg.

If you use a mix of upper and lower case, you have to be extremely consistent.

If you move from a case insensitive, to a case sensitive server, even small errors will break your web.

To avoid these problems, always use lower case file names (if possible).

---

## File Extensions

HTML files should have a **.html** extension (or **.htm**).

CSS files should have a **.css** extension.

JavaScript files should have a **.js** extension.

---

# Differences Between .htm and .html

There is no difference between the .htm and .html extensions. Both will be treated as HTML by any web browser or web server.

The differences are cultural:

.htm "smells" of early DOS systems where the system limited the extensions to 3 characters.

.html "smells" of Unix operating systems that did not have this limitation.

---

## Technical Differences

When a URL does not specify a filename (like `http://www.w3schools.com/css/`), the server returns a default filename. Common default filenames are `index.html`, `index.htm`, `default.html`, and `default.htm`.

If your server is configured only with "index.html" as default filename, your file must be named "index.html", not "index.htm."

However, servers can be configured with more than one default filename, and normally you can set up as many default filenames as needed.

Anyway, the full extension for HTML files is .html, and there's no reason it should not be used.

## HTML Quick List

[« Previous](#)

[Next Chapter »](#)

HTML Quick List from W3Schools. Print it, fold it, and put it in your pocket.

HTML Basic Document

```
<!DOCTYPE html>
<html>
<head>
<title>Title of document goes here</title>
</head>

<body>
Visible text goes here...
</body>

</html>
```

## Basic Tags

```
<h1>Largest Heading</h1>
<h2> . . . </h2>
<h3> . . . </h3>
<h4> . . . </h4>
<h5> . . . </h5>
<h6>Smallest Heading</h6>
```

```
<p>This is a paragraph.</p>
<br> (line break)
<hr> (horizontal rule)
<!-- This is a comment -->
```

## Formatting

```
<b>Bold text</b>
<code>Computer code</code>
<em>Emphasized text</em>
<i>Italic text</i>
<kbd>Keyboard input</kbd>
<pre>Preformatted text</pre>
<small>Smaller text</small>
<strong>Important text</strong>
```

```
<abbr> (abbreviation)
<address> (contact information)
<bdo> (text direction)
<blockquote> (a section quoted from another source)
<cite> (title of a work)
<del> (deleted text)
<ins> (inserted text)
```

<sub> (subscripted text)  
<sup> (superscripted text)

## Links

Ordinary link: <a href="http://www.example.com/">Link-text goes here</a>  
Image-link: <a href="http://www.example.com/"></a>  
Mailto link: <a href="mailto:webmaster@example.com">Send e-mail</a>

Bookmark:  
<a id="tips">Tips Section</a>  
<a href="#tips">Jump to the Tips Section</a>

## Images



## Styles/Sections

<style type="text/css">  
  h1 {color:red;}  
  p {color:blue;}  
</style>  
  
<div>A block-level section in a document</div>  
<span>An inline section in a document</span>

## Unordered list

<ul>  
  <li>Item</li>  
  <li>Item</li>  
</ul>

## Ordered list

<ol>  
  <li>First item</li>  
  <li>Second item</li>  
</ol>

## Definition list

```
<dl>
  <dt>Item 1</dt>
  <dd>Describe item 1</dd>
  <dt>Item 2</dt>
  <dd>Describe item 2</dd>
</dl>
```

## Tables

```
<table border="1">
  <tr>
    <th>table header</th>
    <th>table header</th>
  </tr>
  <tr>
    <td>table data</td>
    <td>table data</td>
  </tr>
</table>
```

## Iframe

```
<iframe src="demo_iframe.htm"></iframe>
```

## Forms

```
<form action="demo_form.asp" method="post/get">

  <input type="text" name="email" size="40" maxlength="50">
  <input type="password">
  <input type="checkbox" checked="checked">
  <input type="radio" checked="checked">
  <input type="submit" value="Send">
  <input type="reset">
  <input type="hidden">
  <select>
    <option>Apples</option>
    <option selected="selected">Bananas</option>
    <option>Cherries</option>
  </select>
```



```
<textarea name="comment" rows="60" cols="20"></textarea>
```

```
</form>
```

## Entities

&lt; is the same as <

&gt; is the same as >

&#169; is the same as ©

**Source :** [http://www.w3schools.com/html/html\\_quick.asp](http://www.w3schools.com/html/html_quick.asp)