

An HTML Primer

This document provides a concise, introductory lesson in HTML formatting.

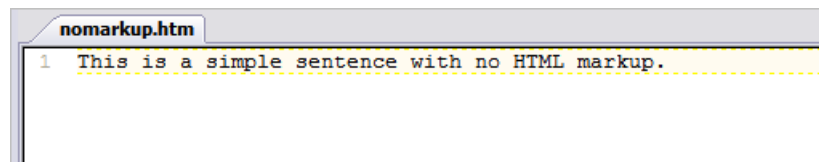
Introduction to HTML

In their simplest form, web pages contain plain text and formatting tags. The formatting tags are a programming or scripting language called *hyper-text markup language* or “HTML.” The files for HTML pages are given the `.htm` or `.html` extensions and are simply known as HTML files.

Browsers know how to “read” and “interpret” HTML files (as well as other formats not covered in this document). If an HTML page does not contain certain formatting tags, browsers can still interpret the pages using default settings and substitutions.

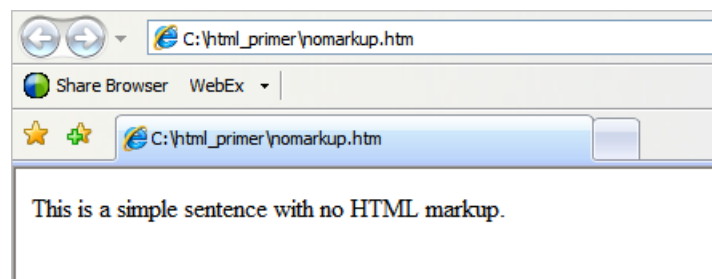
For example, the file shown in [Figure 1](#) is a simple text (`.txt`) file that has its extension changed to `.htm` so a browser can read it. There is no markup whatsoever:

Figure 1: Absolutely no mark—a simple sentence in a text editor



If you open it in Internet Explorer 7, the file in [Figure 1](#) produces the page in [Figure 2](#):

Figure 2: The simple sentence as seen in Internet Explorer 7



The same file as it appears in Firefox ([Figure 3](#)) and Opera ([Figure 4](#)):

Figure 3: The simple sentence as seen in Firefox 3

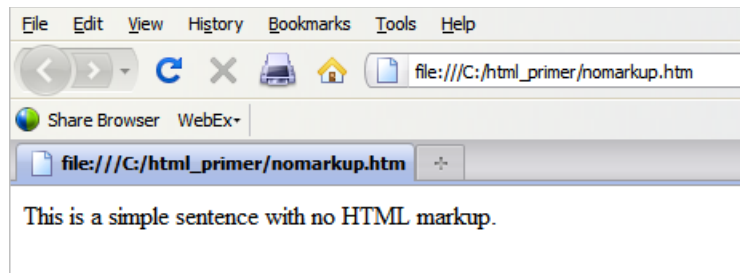
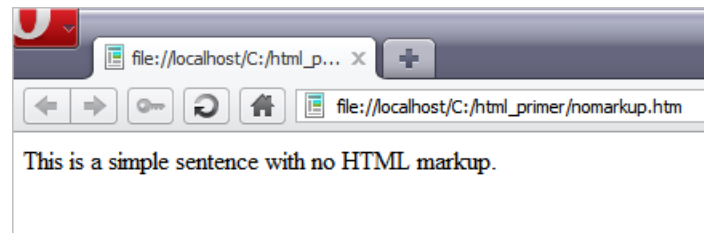


Figure 4: The simple sentence as seen in Opera 10



From now on, we'll just use Internet Explorer 7 for the examples.

HTML Tags

Tags are HTML “code” that dictates how a browser should interpret the associated text. The text is surrounded by a tag. Here is how it works:

- Tags are enclosed in angle brackets: `< >`.
- All tags are matching sets: they include a beginning marker and an end marker. The starting marker turns the feature “on” and the ending marker turns the feature “off.”
- Beginning markers contain an *element*.
- Elements may be followed by an *attribute*, then an equal sign, then a *value*.
- End markers contain a slash followed by the beginning marker’s element.

Let’s suppose, for example, that you want to make some text in a line larger than the surrounding text. Assume the text is set at size 2 (don’t worry about what that means). The original text is shown below:

The big, red dog was a very big dog.

Now you insert a tag to increase the font size of a couple of words:

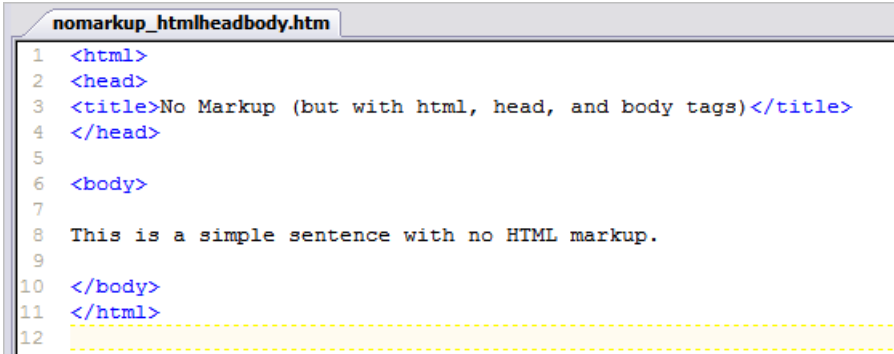
The big, red dog was a ``very big`` dog.

As you can see, the tag is made up of a beginning (`font`) and an end (`/font`). A browser knows that it should do something to the text that is in between those markers; in this example, make the words “very big” size 3.

The element is `font`, the attribute is `size`, and the value is `3`.

Now let's add some important tags for defining an HTML page to the `.htm` file in [Figure 1](#). The next screen shot, [Figure 5](#), shows how it looks in a text editor:

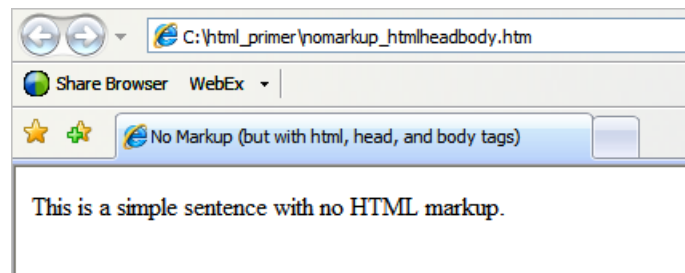
Figure 5: File-defining tags, but still no markup for the sentence



```
1 <html>
2 <head>
3 <title>No Markup (but with html, head, and body tags)</title>
4 </head>
5
6 <body>
7
8 This is a simple sentence with no HTML markup.
9
10 </body>
11 </html>
12
```

(Again, don't worry about what they mean.) The HTML document codes contained in the **head** tag do affect the way the page is displayed, but nothing changes for our sentence because the sentence itself is not tagged. [Figure 6](#) shows you how the page is displayed:

Figure 6: The result as seen in Internet Explorer 7



As you can see, the **title** tag in [Figure 5](#) created a title for the page (compare the tab in [Figure 6](#) with the tab in [Figure 2](#) on page 1).

Danger, Will Robinson! Danger!

In general, you will only work with tags & text that fall between `<body>` and `</body>`.

Do not change any other lines of code—for example, between `<head>` and `</head>`—in the KB articles generated by Technical Publications.

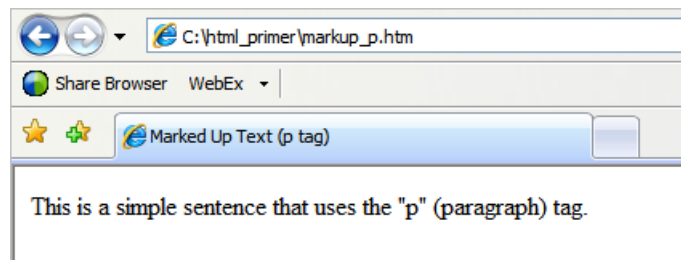
Now let's add a basic tag, the **p** tag (for “paragraph”), to the sentence. [Figure 7](#) shows you the HTML code:

Figure 7: The sentence with a simple "p" tag

```
markup_p.htm
1 <html>
2 <head>
3 <title>Marked Up Text (p tag)</title>
4 </head>
5
6 <body>
7
8 <p>This is a simple sentence that uses the "p" (paragraph) tag.</p>
9
10 </body>
11 </html>
12
```

Figure 8 shows you how the page is displayed:

Figure 8: The result as seen in Internet Explorer 7



Doesn't look much different, right? That's because nothing has been defined to make the page's basic paragraph different from a default browser paragraph.

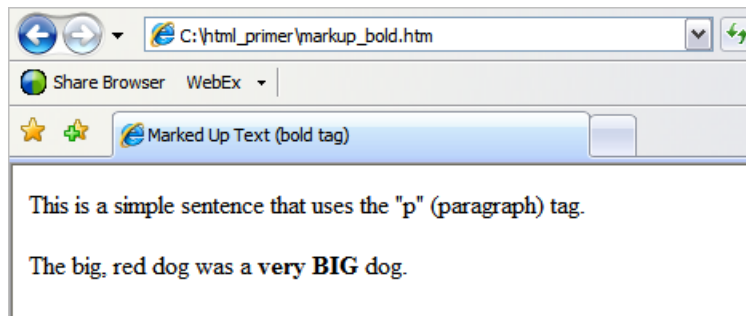
Now let's add a **bold** tag to text within a paragraph in an additional sentence. Figure 9 shows you the HTML code:

Figure 9: Using a bold tag within a paragraph

```
markup_bold.htm
1 <html>
2 <head>
3 <title>Marked Up Text (bold tag)</title>
4 </head>
5
6 <body>
7
8 <p>This is a simple sentence that uses the "p" (paragraph) tag.</p>
9
10 <p>The big, red dog was a <b>very BIG</b> dog.</p>
11
12 </body>
13 </html>
14
```

Figure 10 shows you how the page is displayed:

Figure 10: The result as seen in Internet Explorer 7



Now we'll add a heading to the page and see what the browser does. A first-level heading uses the **h1** tag (for "heading 1"). Figure 11 shows you the added code:

Figure 11: The sentence with an added "h1" tag

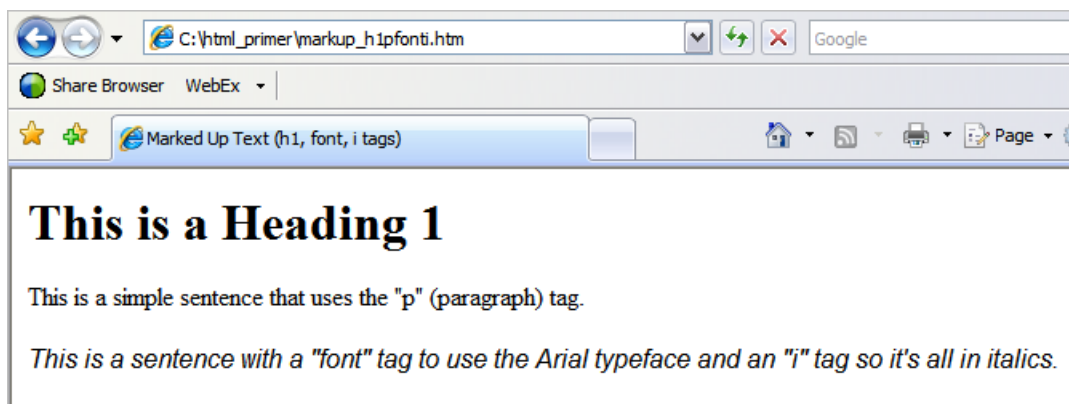
```

markup_h1pfonti.htm
1 <html>
2 <head>
3 <title>Marked Up Text (h1, font, i tags)</title>
4 </head>
5
6 <body>
7
8 <h1>This is a Heading 1</h1>
9
10 <p>This is a simple sentence that uses the "p" (paragraph) tag.</p>
11
12 <p><font face="Arial"><i>This is a sentence with a "font" tag to use the Arial
13 typeface and an "i" tag so it's all in italics.</i></font></p>
14
15 </body>
16 </html>

```

Figure 12 shows you how the page is displayed:

Figure 12: The result as seen in Internet Explorer 7



As you can see in Figure 12, a browser has default formatting for an **h1** tag that is different from the **p** tag's default formatting.

Also, the **font** and **i** tags affect the entire look of the second sentence.

Let There Be Space

Empty lines in an HTML file are not interpreted by browsers as extra space between paragraphs. Use extra lines between your lines of code to make them more readable; they won't add empty space on the web page.

The list below contains some of the basic and most useful HTML tags. Remember that each one is a beginning marker; always close them with the matching ending marker:

- `<h1>`, `<h2>`, `<h3>` . . . `<h6>`: Use for up to six levels of headings.
- `<p>`: Use for paragraphs, otherwise known as body text.

The `p` tag is a similar concept to the “Normal” style in Microsoft Word.

- ``: Use for bold text.
- `<i>`: Use for italicized text.
- `
`: Use to force a line break within a paragraph.
- ``: Use to enclose an unordered list.

List items within this tag have bullets.

- ``: Use to enclose an ordered list.

List items within this tag have numbers or letters.

- ``: Use for a list item within a list, whether it's a `ul` or `ol` list.
- `<hr>`: Use to draw a horizontal line across the page.

Cascading Style Sheets

As you can see in [Figure 12](#), you can change the way entire paragraphs look in a browser by using formatting tags. But what if you had several pages? What if you wanted to change the way an `h2` heading looked on each of those pages? You would have to edit every single page. You'd also have to ensure every `h2` heading on every page has the same formatting tags in order to be consistent.

The problem of changing so many tags is solved by a cascading style sheet (CSS). The style sheet contains the attributes that affect formatting; the pages contain the data (your text with paragraph and list tags only). For example, if you want to change the way an `h2` tag looks in all of your pages, you modify the CSS file only—not every single page.

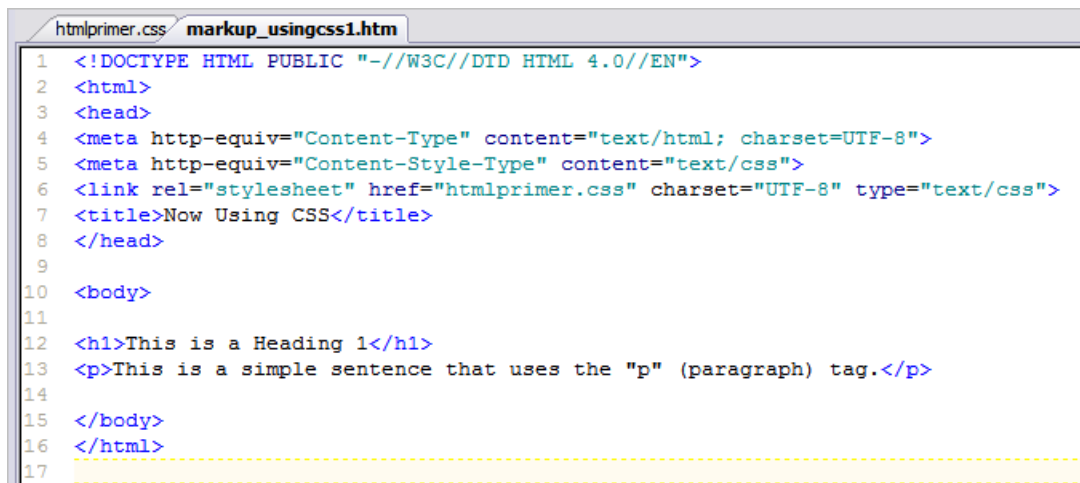
The tags you use when you associate your page to a cascading style sheet (CSS) are not “different” from regular HTML tags; they just use attributes you cannot use with simple HTML.

Note

This document does not cover how to create a style sheet. One has already been created for the Technical Publications KB articles. But, you need to recognize and use some of the tags & attributes associated with the style sheet. Now back to our program . . .

Continuing with our example, let's add some new tags to the top of our page. In particular, the `link` tag on line 6 of Figure 13 ties this page to a CSS file that affects the formatting of this page:

Figure 13: Attaching the page to a cascading style sheet



```
htmlprimer.css markup_usingcss1.htm
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
5 <meta http-equiv="Content-Style-Type" content="text/css">
6 <link rel="stylesheet" href="htmlprimer.css" charset="UTF-8" type="text/css">
7 <title>Now Using CSS</title>
8 </head>
9
10 <body>
11
12 <h1>This is a Heading 1</h1>
13 <p>This is a simple sentence that uses the "p" (paragraph) tag.</p>
14
15 </body>
16 </html>
17
```

Figure 14 shows you how the page is displayed:

Figure 14: The result as seen in Internet Explorer 7



As you can see in [Figure 14](#), the style sheet has changed the look of the page in [Figure 12](#) on page 5:

- The **h1** heading is centered, is a different color, and uses a different font.
- The **p** paragraph uses a different font.
- The background color of the page is different.

All these changes were a result of associating the page to the style sheet. Modifications to the CSS automatically and immediately affects all its associated pages.

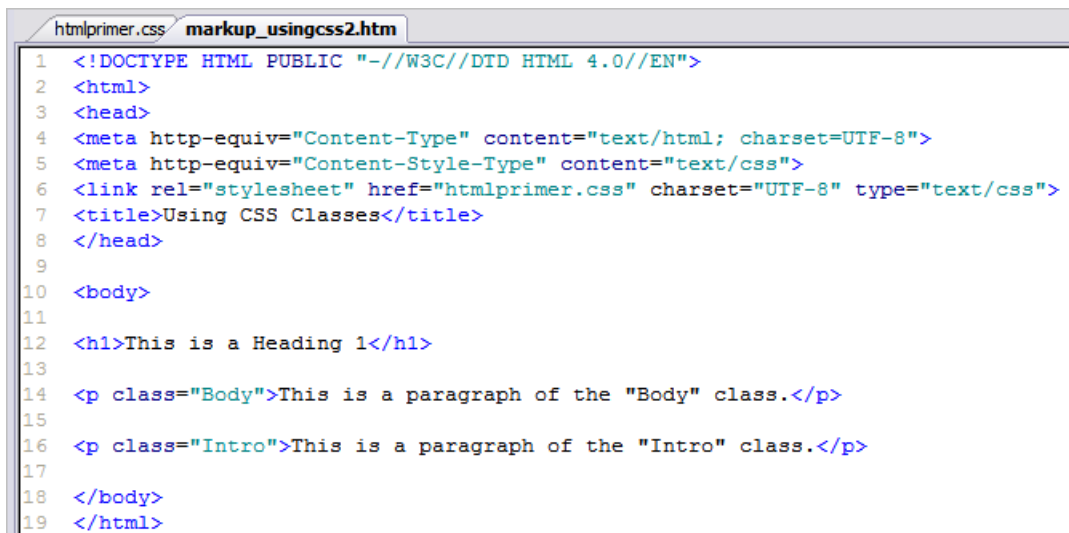
CSS Classes

Because of the formatting definitions in the style sheet, you use a different attribute in the tags on your HTML page. It is the **class** attribute.

The **class** attribute allows you to define different types of paragraphs. You can have two types of h1 headings, or 10 different types of p paragraphs. Each type of paragraph is named in the style sheet; each **class** attribute indicates the name in the HTML page.

Continuing with our example, let's add the **class** attribute to a couple of paragraphs. We reference two different types of paragraphs, as shown in [Figure 15](#):

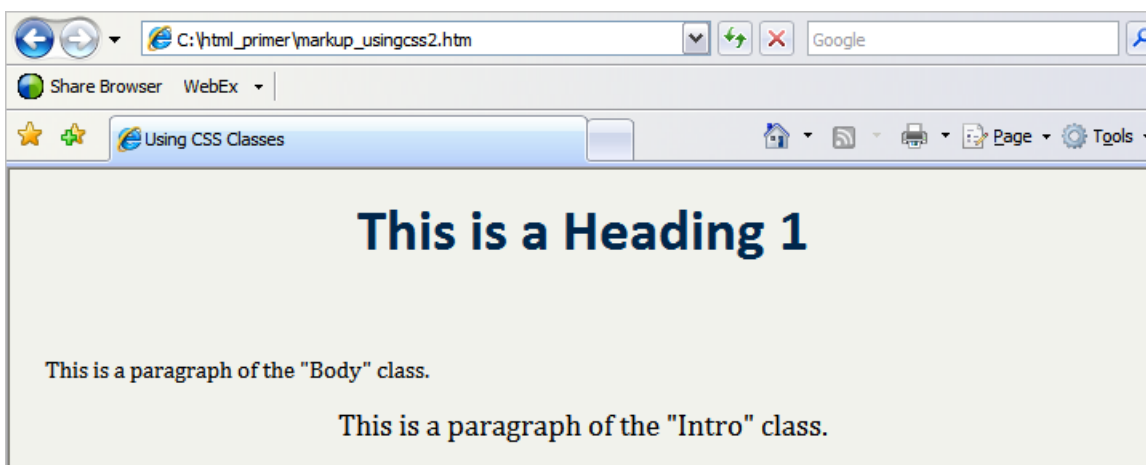
Figure 15: Adding the class attribute to paragraph tags

A screenshot of a code editor window. The window has two tabs: 'htmlprimer.css' and 'markup_usingcss2.htm'. The code is as follows:

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
5 <meta http-equiv="Content-Style-Type" content="text/css">
6 <link rel="stylesheet" href="htmlprimer.css" charset="UTF-8" type="text/css">
7 <title>Using CSS Classes</title>
8 </head>
9
10 <body>
11
12 <h1>This is a Heading 1</h1>
13
14 <p class="Body">This is a paragraph of the "Body" class.</p>
15
16 <p class="Intro">This is a paragraph of the "Intro" class.</p>
17
18 </body>
19 </html>
```

Figure 16 shows you how the page is displayed:

Figure 16: The result as seen in Internet Explorer 7



Formatting (text color, italics, and so on) is also different when using CSS. The details of using a cascading style sheet are explained in an upcoming tip sheet.

