Changing the look and feel of text
Lesson 7

The Web is a text-based environment, with some graphics and other multimedia files thrown in for good measure. If you take a few minutes to surf the Web, you will find highly stylized Web pages that include text of every font type, size, and color. Some text is bold, while other text is italicized or underlined. Yet the page you've created in this course doesn't include any of those nuances.

If you think that there are HTML tags that you can use to control the look and feel of your text, you're right. But before you go on to learn about those tags, think about the consequences of being able to control the color, typeface, and size of every bit of text on your page.

Different browsers and different displays

Have you ever surfed the same Web site with two different Web browsers or with the same brand of Web browser on two different kinds of computers? If so, you probably noticed that the site didn't look exactly the same from browser to browser or platform to platform. This is not necessarily a bad thing.

Remember that the Web is text-based, so that anyone can view its content on any computer. The HTML standard drives Web development, so you can build a single Web page and any browser that reads HTML can display that page. This flexible system means that your Web pages can be seen around the globe by millions of people using a host of different browser configurations. This is a clear benefit: You have to create only one Web page and all those people can see it.

Of course, with so many different Web browsers floating around, not every person who views your Web page will see it in exactly the same way. Your content will appear, and although differences among browser displays may be slight, there are differences just the same. Web designers who come from a print background have a hard time accepting that their pages won't look exactly the same to everyone who views them. This one point illustrates how different the Web is from print media. Once you understand and accept that, you're well on your way to being able to take advantage of the reality of the Web.

One of the most popular Web browser features is the ability to control the personal viewing environment. Users can choose which fonts appear, how wide the browser window is, and how large the text appears on screen. For some users, setting up such things is simply a matter of personal preference; for others,

Changing your view

It doesn't matter how good your video card is or how great your web page looks if your old monitor displays it badly. Upgrade to an HP monitor that will bring out the best in your work.

» HP L1940T flat panel monitor

it is a necessity. When you apply styles and colors to your page remember that there is a small chance a user's browser preferences will override them. If your content is compelling this is OK, the style and colors just add pizzazz, they don't take the place of what your page has to say.

The importance of accessibility

Any device that can read HTML can display a Web page, which makes it possible for users with disabilities to surf the Web. Users with visual disabilities can configure Web browsers to display text in very large type and use specific color combinations to make the pages more viewable. Text-to-speech readers can read Web pages aloud and special peripheral devices can output Web pages in Braille.

The more tinkering you do with the look and feel of your Web pages, the less accessible they are. To be sure that your pages work in any environment, turn off the graphic display after adding tags to control page design and set your preferences to override page font settings. (Both options are available in your browser options or preferences.)

Web page accessibility is so important that the W3C has developed a series of guidelines for Web page developers. You can -- and should -- read more about them at http://www.w3.org/WAI/. If you can get into the habit of following these guidelines and testing your pages for accessibility, you'll develop good page design habits from the start.

Plain text is simple and to the point, but it doesn't always convey the message that you want it to and it doesn't provide much in the way of emphasizing important points. To that end, HTML has a variety of elements that add emphasis and contrast to different parts of your text. Each of these is a tag pair: the open tag turns the text formatting on and the close tag turns the formatting off. If you forget the close tag, the browser applies the formatting to everything in your document from the open tag to the end of the document.

Your options for adding emphasis and contrast to text in your document include

      **\ . . . **: boldface
      \<i> . . . \</i>: italics
      \<code> . . . \</code>: code (displayed in a monospace font)
      \<big> . . . \</big>: bigger text
      \<small> . . . \</small>: smaller text
      _{. . . \}: subscript
      \^{. . . \}: superscript

Protecting your data

You never know when a power outage, system malfunction, or other disaster will do away with your important data. Recreating even a single web page can take more time than you have, so take the time now to plan for disaster recovery.

» HP Care Pack services

The following code uses all of these formatting elements and Figure 7-1 shows how a Web browser interprets the code.

```
<p>This text is displayed in
<b>boldface</b>.</p>

<p>This text is displayed in
<i>italics</i>.</p>

<p><code>This text is displayed as
code.</code></p>

<p>One word in this sentence is
<big>bigger</big>
    than the others.</p>

<p>One word in this sentence is
<small>smaller</small>
    than the others.</p>

<p>Water can be described as
H<sub>2</sub>0.</p>

<p>Old M<sup>c</sup>Donald had a farm.</p>
```
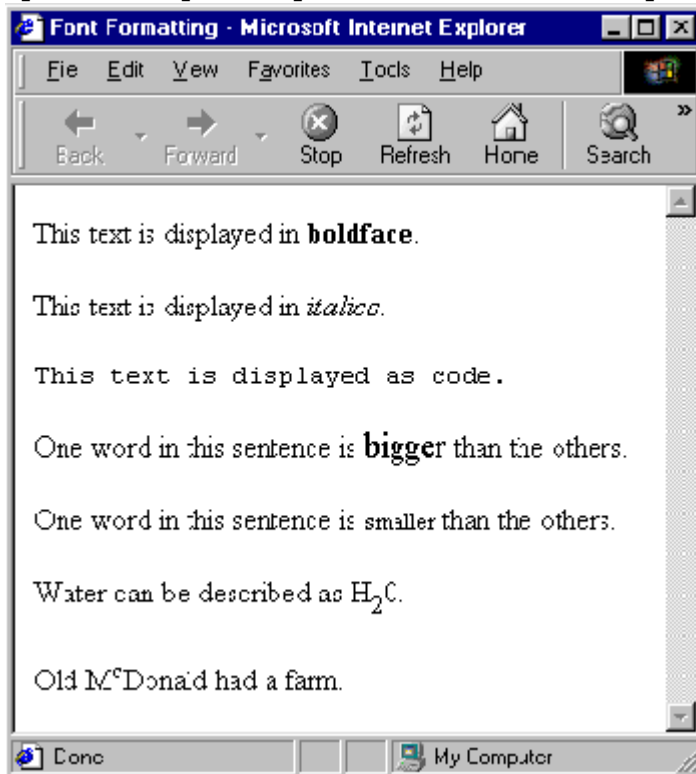

Figure 7-1: This is how a Web browser displays the formatting tags used in the code above.

You can combine these tags to create text that is both italicized and boldfaced or to create boldfaced code or code with subscript and superscript text in it. Keep the following things in mind when

you combine formatting tags:

Too much formatting can make your text difficult to read. Choose one formatting tag to emphasize text or set it apart from the text around it.
When you combine formatting tags, be sure to nest your tags correctly. Always close last what you opened first.

Nesting tags

HTML tags that apply formatting to text are called *inline tags*. That is, they apply to bits and pieces of text that are part of a block, rather than describing whole blocks of text like paragraphs and headings do. For this reason, you should use inline tags within blocks already described with a block tag, like a paragraph or heading. If you use these tags to format a whole paragraph or series of paragraphs, be sure to nest the formatting tag within the paragraph tag, like this:

```
<p><b>This entire paragraph should be displayed in
   boldface.</b></p>
```

Most Web browsers won't do strange things to your Web pages if you flip-flop the bold and paragraph tags in this example, but it is still bad code and you want to avoid bad code whenever possible.

There are two easy ways to add color to your text:

Set a single color for all of the text on your page.
Apply color to specific lines of text on the page.

Setting color for the whole page

To set a single color for all of the text on a page you use the `text=` attribute with an element you've seen before, `<body>  .  .  . </body>`. It works like this:

```
<body text="teal">
the rest of your document goes here
</body>
```

The `text=` attribute applies to all of the text in your document because it is set in the `<body>` tag at the top of your document. This is a really simple way to color all the text on your page. You might want to think twice before you make it all lime green, but hey -- it's your Web page!

Beyond HTML for style

Chances are that you want to use color to highlight page elements of a particular type, like headings or titles. As you've seen, this is tedious when you use the HTML `font` element. There is an easier and better way to set font color for your document, but a complete discussion of it is outside of the scope of this course. Cascading Style Sheets (CSS) allow you to define more carefully how any given tag should display on the page. You can read more about CSS in the Web Design Group's CSS Guide.

Setting color for specific lines of text

The judicious use of different text colors can add highlights and visual appeal to a Web page. Differently colored headings stand out on a page. Newsflashes or special notes displayed in a different color draw a user's attention. Use this element-and-attribute pair to change the font color for a specific section of text:

```
<font color="teal">This text will be teal</font>
```

Web browsers display everything in between the `<font>` and `</font>` tags in the color you specify with the `color=` attribute. In this example the sentence in the middle of a paragraph is teal while the text around it is the standard black. Figure 7-2 shows how a Web browser interprets the following content:

```
<p>This is a regular old paragraph. Notice that this
text is displayed in the standard font color.
<font color="teal">This text, however, is displayed in
a teal font.</font>
Now that the font tag is closed the text is back to
plain ol' black.</p>
```
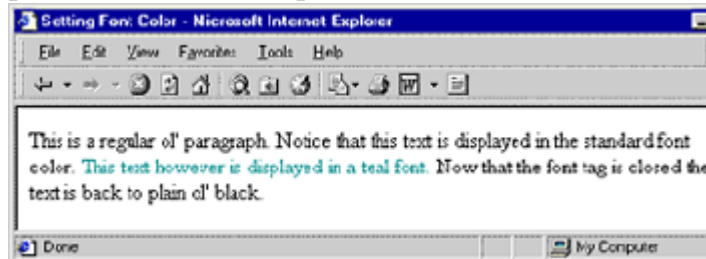

Figure 7-2: Use the font tag to apply color to lines of text in a document.

Drawbacks of this method

One of the major drawbacks of using the `font` element is that you have to set colors in your Web page one line or section of text at a time. For example, if you want all of your first-level headings to be maroon, you have to apply a font element to each and every heading, as in this example:

```
<h1><font color="maroon">
 This is a first level heading
</font></h1>
<h1><font color="maroon">
 This is also first level heading
</font></h1>
<h1><font color="maroon">
 This is another first level heading
</font></h1>
<h1><font color="maroon">
 This is a fourth first level heading
```

```
</font></h1>
```

There is no easy way in HTML proper to specify that all of your first-level headings should be displayed in maroon text (see the sidebar for a non-HTML way). If you take a peek at the code for any Web page that uses lots of text coloring, you'll see `font` elements scattered liberally around the page. A good HTML editor such as BBEdit or HomeSite helps make this process a bit easier because they have shortcut buttons and other tools that let you set font color more quickly -- and with less chance for error -- than you could by typing the code yourself.

Let's suppose you want the majority of your text displayed in one color and to apply font settings to selected other spots in your document. Use the `text=` attribute with the `body` element to set your base color, then use the `font` element to set other colors in the document.

The font element is an inline tag, just like the boldface, italics, and other formatting tags. This means that you don't want to use the font tag alone to describe a block of text. Instead, your font tag should be nested within a block element such as a paragraph or heading.

Naming colors   Page 4 of 6

In the examples on the previous page I used teal and maroon as valid values for setting color in a Web page. There are several different ways to specify color, including color names and a special color-coding system called hexadecimal color notation. You can get started using colors in your Web pages by choosing your colors from this list:

> Black
> Silver
> Gray
> White
> Maroon
> Red
> Purple
> Fuchsia
> Green
> Lime
> Olive
> Yellow
> Navy
> Blue
> Teal
> Aqua

Most Web browsers recognize these 16 color names, so you can use them to set color in the assignment for this lesson and as you experiment with your own Web pages. Figure 7-3 shows how a Web browser displays the different colors.

Beyond the standard colors

If you want to use a color on your Web page that isn't listed here you're in luck. The hexadecimal numbering system lets you express just about any color on your page. Good HTML editors like BBEdit and HomeSite have color picking tools built in that you can use to find the code for your favorite color. If you know a lot about how color works and want to create the code yourself you can read this interesting article on the subject. Finally, it's important to know that some colors work better across Web browsers than others. You should always try to use colors from the Web safe color pallet. You can read all about Web safe color on Lynda Weinman's site. This modern maven of
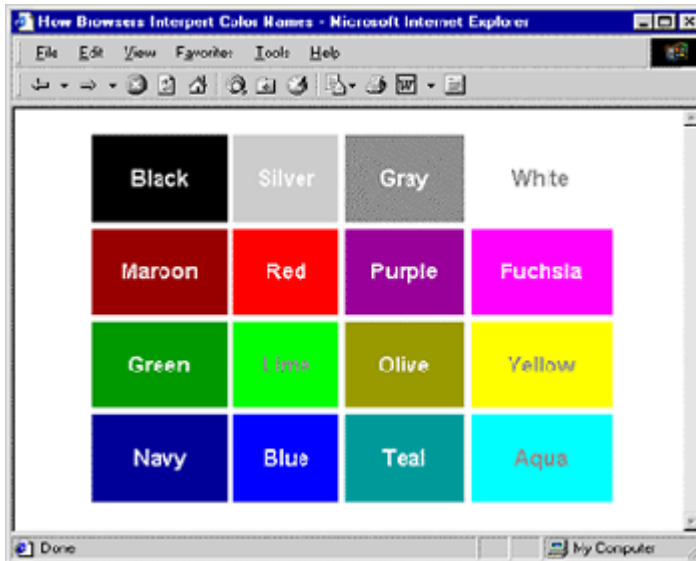
Figure 7-3: A Web browser display of the 16 basic colors.

Web browsers display text in the default font face that users set in their preferences. You can change that font face or use more than one font face in your page; for example, Arial for headings and Garamond for text. In addition to controlling font color, the `<font>` element also controls font face, as in this code:

```
<font face="Arial">This text will be displayed in
   Arial type.</font>
```

It's really that easy. You can combine the `color=` and `face=` attributes in a single font element to set both color and font face, as in this example:

```
<h1><font face="Arial" color="maroon">

This first level heading will be displayed in
maroon
Arial type.

</font></h1>
```

Figure 7-4 shows how a browser renders this bit of code.

A shortcut that isn't really a shortcut

You may be tempted to place one `font` element around your entire HTML document to set the font face instead of using the font tag with each paragraph or heading. Many HTML browsers will even interpret your font tag exactly as you'd like and apply the font tag settings to the entire document. While this may seem like a good shortcut to avoid much typing of tags and attributes, it isn't really a shortcut.

This particular technique is in violation of good HTML coding practices and doesn't work in all browsers, so you won't get a consistent display from browser to browser. It's better to take the time to set individual font tags for all of the elements in
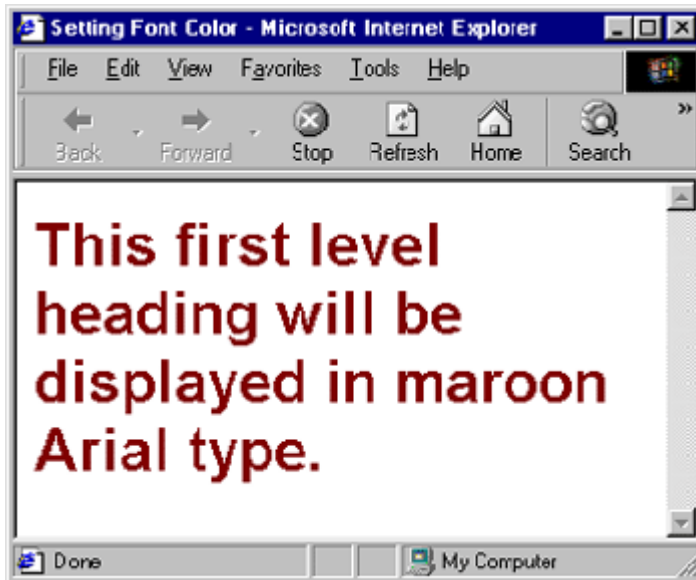
color can teach you everything you need to know about the common sense of color.

Figure 7-4: Use the font tag to set both typeface and text color.

your document than to use a shortcut that doesn't really get you anywhere.

There are a few things you should bear in mind when you use the font tag to set typeface:

The page will be displayed in the typeface you specify only if that typeface is loaded onto the user's computer. If the typeface isn't available to the Web browser, the browser ignores your type specification. To be on the safe side, stick to fonts such as Arial, Times, Geneva, Helvetica, and Courier that are common to a variety of platforms.

Too many typefaces or strange typefaces can make it difficult for users to read your content. Keep your font usage down to two or possibly three different fonts on a single page. Common usage is to set headings in one font, document text in a second font, and special items in a third if the page is particularly long and complex.

Users can turn off Web page font settings in their browsers, effectively ignoring your font tags. Don't take it personally. They can still read your content and surf your site.

The way you apply font settings to a Web page can dramatically change its overall look and feel. When you finish the step-by-step exercise in the next section, the ACME Widgets page will have had a makeover. Cosmetic changes are often the last work you'll do on a Web page, so it's fitting that when you finish working through the next section you'll be finished building the ACME Widgets Web page.

Moving on   Page 6 of 6

Playing with the font face, color, and size for the different bits of text on your page can be fun. The assignment for this lesson gives you a change to spruce up the ACME Widgets Web page with some color and font face variety. Try it, you'll like it.

With this lesson we come to the end of the discussion of how to use HTML tags in your Web page. As

you might have noticed, this class doesn't cover all of the HTML tags out there, but there should be enough to whet your appetite. The course text includes a complete discussion of more advanced HTML topics, including tables, frames, and forms. When you are comfortable with basic HTML, you should move on to these fun and very useful topics. A great online tutorial for advanced HTML is at Webmonkey.

The last lesson of the course helps you put your new Web page up on the Web for the world to see. You'll learn about the variety of options for hosting your Web page, and you'll even have the chance to post your page. After all, if you can't share your Web page then why go to the trouble of building it?