

Topic Notes: Style Attributes

The HTML we have written so far gives us some control over presentation, but you have probably found the resulting pages pretty boring. Most of our HTML has been to specify document structure, and we're leaving many of the details of presentation to the browser.

We will look at a number of presentational attributes, often called style attributes, that will give us much more control over the appearance of our documents. There are many ways to specify these – we will start by adding some of them to the HTML tags we already know.

Generally, we will add a `style=` attribute to the opening tag of an element to have the style applied to the text within the element.

The general format:

```
<p style="property:value"> ... </p>
```

Font-related Styles

We begin by specifying styles that will change the intended font of an element. We will use paragraph elements to start, but these can be applied to nearly any element.

- `font-style` is most commonly used to control italicized text, and has possible values of `normal` (the default), `italic`, or `oblique`.
- `font-weight` controls the weight (“boldness”) of the font. It has many possible values, but the most common are `normal` (the default) and `bold`.
- `font-size` specifies the size of a font. It can be specified in a number of ways, some of which are more appropriate than others for a given situation.
 - Absolute sizes can be specified as a point size like `12pt` but this is not usually appropriate for all possible browsers.
 - Better options for absolute sizes are the values `xx-small`, `x-small`, `small`, `medium` (the default), `large`, `x-large`, and `xx-large`.
 - Relative sizes can also be specified: `smaller`, `larger` to choose a font a little smaller or a little larger than the font that would otherwise be used.
 - A more precise relative size can be achieved using a percentage, where values under 100% reduce the size, greater than 100% increase.

- `font-family` specifies a font family such as `Sans-serif`, `Monospace`, `Times New Roman`, or even something less common like `cursive`.
-

Color

A very common style property is to change the color of the text in an element.

The `color` property unsurprisingly specifies element colors.

There are two main ways we can specify the colors: by name, or by *RGB* (*red/green/blue intensity*) value.

The names understood by this style are very limited. There are only 16 that you can count on: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow. You can try others, but they may not work on all browsers. While such a small set of named colors may be good enough for many purposes, there are nearly 2^{24} other colors we can specify that do not have names.

For these, we specify how to create a color by combining a certain amount of red, green, and blue light, the primary colors of light. This is exactly what most of your CRT, LCD, and other displays do.

The most common way to specify RGB colors in a style is using a *hexadecimal* or base 16 notation. The values for each color range from 0 through 255, which in hexadecimal is from `0x00` to `0xff`. By specifying 2 hex digits for each color, we can specify a color in 6 hex digits.

For example, `color="#ffff00"` would generate yellow, since it is a combination of as much red and green as we can get, and no blue. (Yes, in light, red+green makes yellow).

Alternate specifications:

- `#9d0`: 3 hex digits specifying the red/green/blue components on a 0-15 (0-f in hex) scale for each instead of the 0-255 (00-ff in hex) scale.
- `rgb(73,200,24)`: 3 decimal numbers specifying the red/green/blue components on a 0-255 scale.
- `rgb(17%,100%,23%)`: 3 percentages specifying the red/green/blue components on a 0%-100% scale.

There are many programs that can be used to visualize the colors specified by different RGB values. I like this one:

http://www.cs.rit.edu/~ncs/color/a_spaces.html

Two color-related properties can be applied to nearly all of the HTML elements we have seen:

- `color`: specify the color of the element

- `background-color`: specify the background color of the element

The “Gaudy Colors” example linked from the lecture page shows a fairly extreme example of using color attributes in a variety of elements.

This example shows that style attributes can be specified at each level in the document hierarchy. Inner elements can override the values specified at outer levels. For example, consider the table, where a color is specified for the entire table. But if a color is specified for a row, that overrides the table color. And if a color is specified for a table cell, that overrides the row or table color.

The `span` Element

We also see an example of a new element, the ` ... ` element. This does not specify any layout or other changes to the visual display of the document on its own, but it does give us the chance to specify some style information that applies only to the part of the document within the element.

Other Style Attributes

We will not look at all of the style properties at once, but we will take a quick look at a list.

On the web: CSS Properties from Web Design Group at
<http://htmlhelp.com/reference/css/properties.html>