Colors

**The Power of Colors**

In the last unit, you learned about the basics of CSS, including:

1. How to link HTML and CSS files together
2. How CSS syntax works

Along the way, we styled a different properties of HTML elements, but the details of each property weren't explained. For example, we changed the colors of headings and paragraphs, but we didn't explore the full power and versatility of color.

Professional websites often use various aspects of color in order to create a visually appealing user experience (UX). In this lesson, you'll learn how to use CSS to style web pages with color.

Let's begin!

**Foreground vs Background**

Before discussing the specifics of color, it's important to make two distinctions about color. Color can affect the following design aspects:

1. The foreground color
2. The background color

Foreground color is the color that an element appears in. For example, when a heading is styled to appear green, the *foreground color* of the heading has been styled.

Conversely, when a heading is styled so that its background appears yellow, the *background color* of the heading has been styled

In CSS, these two design aspects can be styled with the following two properties:

1. color - this property styles an element's foreground color.
2. background-color - this property styles an element's background color.

h1 {

color: Red;

background-color: Blue;

}

In the example above, the text of the heading will appear in red, and the background of the heading will appear blue.

# Named Colors

Over the past few exercises, you've seen CSS examples that use colors like Red, Blue, or Cyan. In CSS, these colors are technically known as named colors. There are a total of [147 named colors](http://www.colors.commutercreative.com/grid/).

At this point, you might be wondering if you are expected to memorize the list of 147 named colors that CSS offers.

Fortunately, you don't have to! There are plenty of available resources on the Internet that list all of the named colors in CSS, like the one we linked you to above. If you're ever in need of a named color, a quick Google search will yield the results you are looking for.

Let's try some new named colors out!

# RGB Colors

Although named colors provide 147 different options, this is a small amount when you consider the flexibility of CSS. To take advantage of the full spectrum of colors that CSS supports, you have the option of using RGB colors.

RGB (Red, Green, Blue) colors offer the option of 16,777,216 possible colors. How is that possible?

RGB colors work by mixing together different amounts of red (R), green (G), and blue (B). Each color (R, G, or B) can take on 1 of a possible 256 values (between 0 and 255). This results in 16,777,216 possible colors.

To use RGB colors, you can use the rgb() value when styling a color.

h1 {

color: rgb(123, 20, 233);

background-color: rgb(99, 21, 127);

}

In the example above, the value of color is set to rgb(). The three numbers in the parentheses represent the values for R, G, and B, in that order. Note that you can use rgb() for background colors as well.

How can you tell what color the RGB values in the example above will result in? Are you expected to memorize 16,777,216 possibilities? Thankfully, no!

There are many resources on the Internet known as "[color pickers](https://color.adobe.com/create/color-wheel/)" that allow you to view the result of different RGB values before you decide to use a certain color. If you're ever in need of a color picker resource, a quick Google search will yield the results you are looking for.

# Hex Colors

There's an additional way to specify colors in CSS: hexadecimal color codes, often referred to as "hex color codes" for short.

Hex color codes also offer 16,777,216 color options, but they follow a different syntax.

When specifying an RGB color mixture, the values are in [base 10](https://en.wikipedia.org/wiki/Decimal). Hex color codes, however, use [base 16](https://en.wikipedia.org/wiki/Hexadecimal), or hexadecimal base (hence the name), to specify color mixtures.

h1 {

color: #09AA34;

}

When read from left to right, each group of two characters responds to a value for red, green and blue, respectively. In the example above, 09 refers to the value for red, AA refers to the value for green, and 34 refers to the value for blue. All hex color codes begin with a # character.

Is there a difference between RGB values and hex color codes?

Not really. RGB values and hex color codes are different ways to represent the same thing: color. It's possible to convert back and forth between RGB values and hex color codes (color pickers often help with this conversion).

**Note:** When a hex color code is composed of entirely of the same characters, the hex color can be abbreviated, like so:

h1 {

color: #FFFFFF;

color: #FFF; /\* This is the same color as above \*/

}

h2 {

color: #AA33BB;

color: #A3B; /\* This is the same color as above \*/

}

# HSL

The current revision of CSS, CSS3 (at the time of this writing), introduces a new way to specify colors using HSL colors.

HSL stands for **H**ue, **S**aturation, and **L**ightness. Specifically, this is what each means:

1. Hue - the technical term that describes what we understand as "color." In HSL, hue is represented on a color wheel. It can take on values between 0 and 360.
2. Saturation - the amount of gray in a given color. In HSL, saturation is specified using a percentage between 0% and 100%. The percentage 0% represents a shade of gray, whereas 100% represents full saturation.
3. Lightness - the amount of white in a given color. Similar to saturation, lightness is specified using a percentage between 0% and 100%. The percentage 0% represents black, whereas 100% represents white. 50% is normal.

You can use HSL colors in your CSS like this:

h1 {

color: hsl(182, 20%, 50%);

}

Notice that using HSL is very similar to using RGB.

**Note:** Because HSL is a part of CSS3, older browsers may not support it. In a later exercise, you'll learn how to work around support issues for colors.

# The Alpha Value: a

You learned that RGB and hex color codes are two different methods of representing the same thing: color. However, there is one feature that RGB colors support that hex color codes do not: opacity.

Opacity is a measure of how transparent a color is. To modify opacity in RGB colors, CSS offers the rgba() value. Note the slight difference in rgb() and rgba().

The extra a character in the rgba() value is known as the alpha value. It represents the opacity of a color. The alpha value can be a number between 0 or 1, inclusive.

h1 {

color: rgba(123, 88, 9, 0.5);

}

In the example above, the alpha value has been set to 0.5. This indicates that the color of the heading will be set to 50% of its normal opacity.

**Note:** The alpha value can also be used for HSL colors, using hsla():

h1 {

color: hsla(239, 45%, 22%, 0.4);

}

# Color Declarations

RGB colors, hex color codes, and HSL colors offer web developers an extraordinary amount of color customization options. As these properties become more advanced, however, it's important to keep in mind that not all users browse the Internet with the same browser, let alone the same version of a given browser.

How does this affect web development? Newer revisions of HTML and CSS affect older browsers. Older browsers, over time, will become dated (possibly obsolete) and not be able to support newer CSS features. For example, many older browsers do not support RGBa, HSL, or HSLa.

Because of this, we must include redundant color options in our CSS code that can cater to a wide audience of different browsers.

Specifically, we can add multiple CSS color declarations, just in case a user's browser can't support a certain declaration.

h1 {

color: rgb(22, 34, 88);

color: rgba(22, 34, 88, 0.4);

}

In CSS, the latter of multiple declarations takes priority. In the example above, if the user's browser supports rgba(), then that color will be applied to the heading. If it does not, then CSS will use the first rgb() color declaration, as a backup.

Using redundant declarations allow you to support as many users as possible across multiple versions of different Internet browsers.