Making sense of Colors in CSS

Colors make up beautiful designs - in most cases. There's not going to be a lot of color theory in this lesson, but you will learn to manipulate colors with CSS. You'll also learn the color units in CSS. Pretty much all the important CSS color stuff.

We will begin this lesson with a quick example. Take a look at the code below:

```
body {
  background-color: red;
}
```

valid color names

The code above shows how we have handled colors from the start of the course. We have used names such as `red`, `blue`, `green` etc.
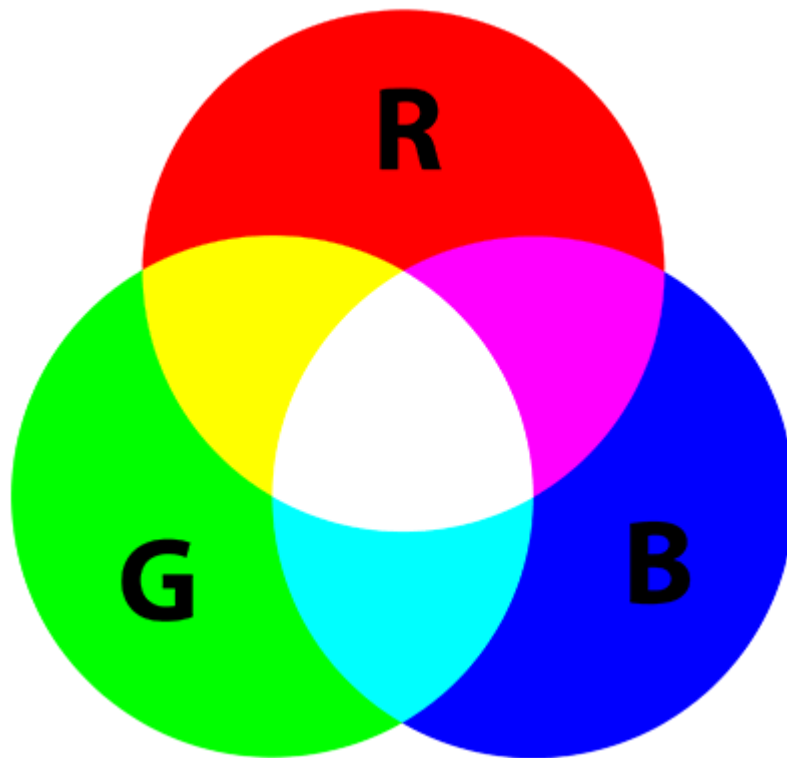
These color names are generally referred to as `color names` and there are about 140 of these names - all valid. You can take a look at all of them here

Before jumping off to check out these colors, don't try to memorize the colors. Just take a look at them 👆

I know 140 color names feel like a lot, even then, web designers and developers need more! Let's take a look at the other ways colors may be defined in CSS.

1. RGB values

Take a look at the image below. What does it remind you of?

For me, it does remind me of when I was a kid. When I toyed with colored pencils. ✏️

The important thing to note is the MIX of colors. 3 primary colors (red, green, and blue) mixed together to produce various other colors.

This concept is equally adopted in CSS too. Only, instead of mixing these colors with pencils, we will do so in values from `0 - 255`.

Examples

Let's have a look at some examples.

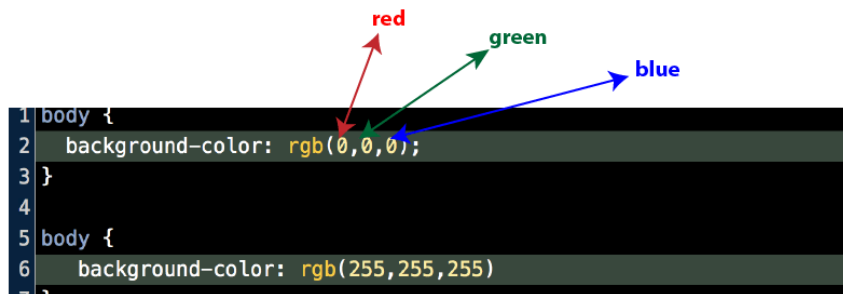In its most basic form, colors are defined in `rgb` values as seen below:

```
body {
  background-color: rgb(0,0,0);
}

body {
    background-color: rgb(255,255,255)
}
```
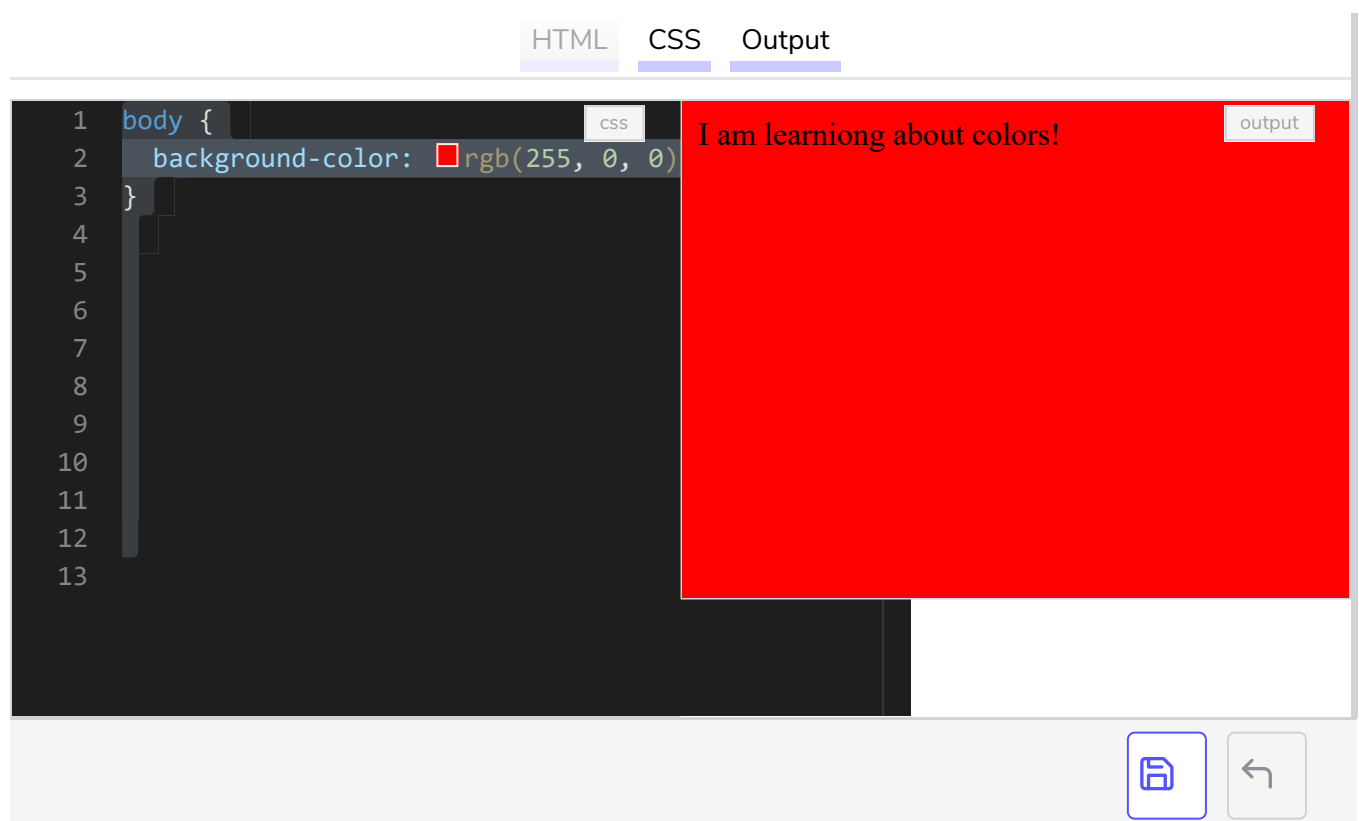
basic color declarations

Now we have taken away the color names and replaced them with `rgb()`

In case you were wondering, the first value within the rgb() declaration is assigned to the `red` color, and the other other two to, `green` and `blue` respectively.
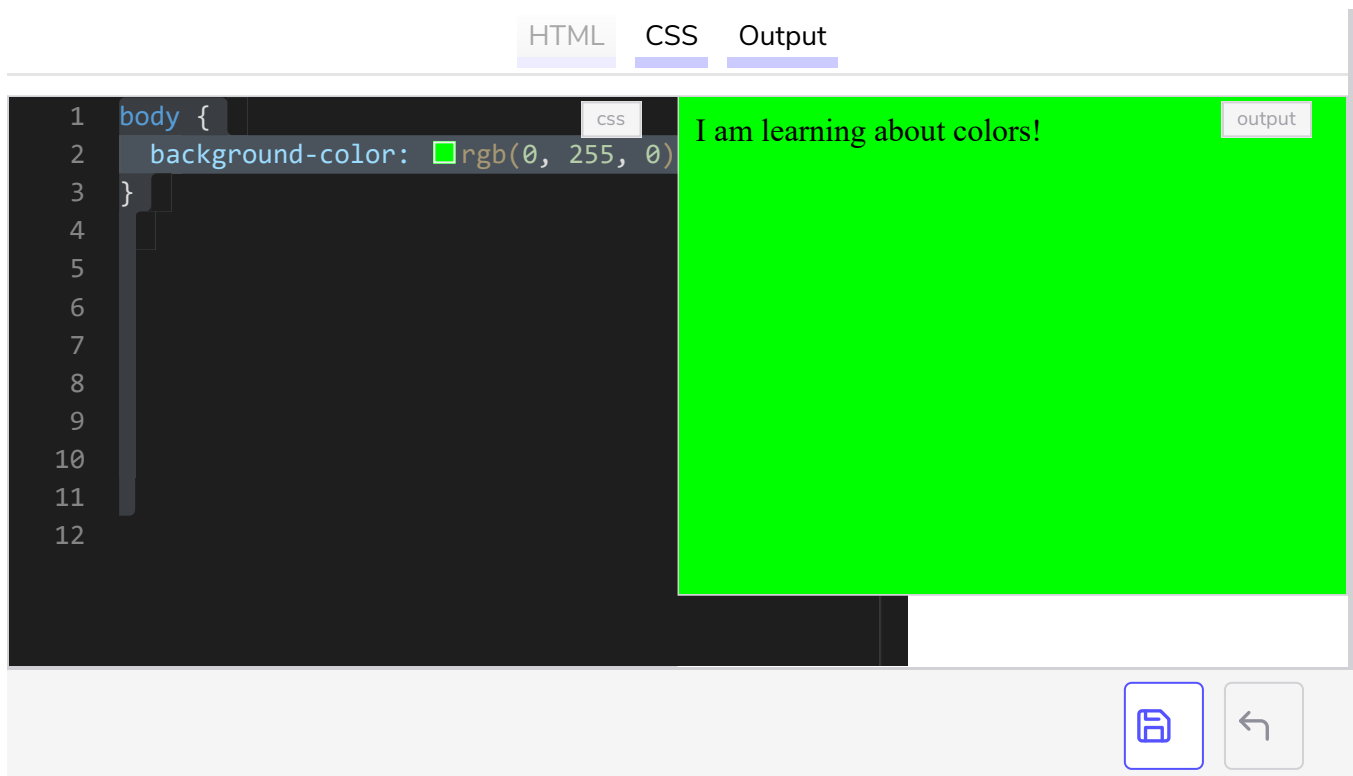

Here are some more practical examples.

1. `rgb(255,0,0)` says, fill the *"color bucket"* with `255` bowls of red paint, and NOTHING for green and blue. Thus, this yields a fully red color

Please see the playground above.

2. Setting the background color of the page to `rgb(0, 255, 0)` **yields what?**

"No buckets of Red, 255 buckets of green, and no buckets of blue too"


```css
1  body {
2    background-color: ▮rgb(0, 255, 0)
3  }
4
5
6
7
8
9
10
11
12
```
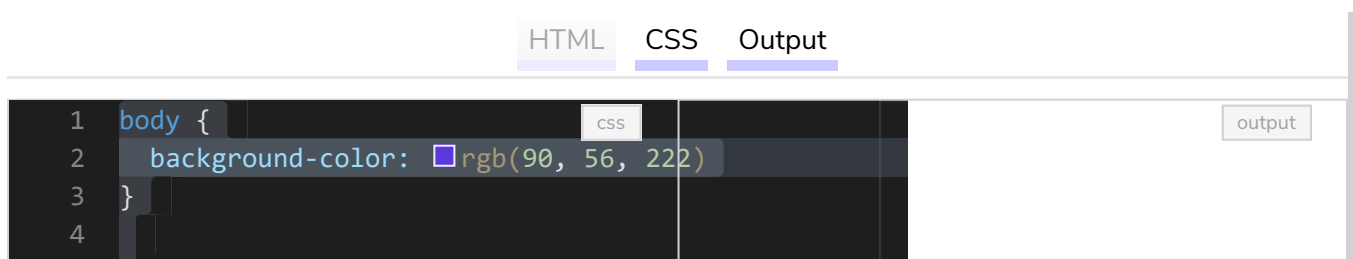
I am learning about colors!

Like before, click the "output" tab to see the result.

3. Now I think you've gotten a hang of it. So go ahead and set the background-color of the page in the code above to `rgb(0,0,255)`. **You should get a blue page then.**

More color options

There are certainly more combinations asides blue, green and red. The three colors can be mixed in very different ways. From 0 - 255. That means you can have colors such as `rgb(0,34,127), rgb(89, 241, 8) or rgb(90, 56, 222).` The possibilities are really much.
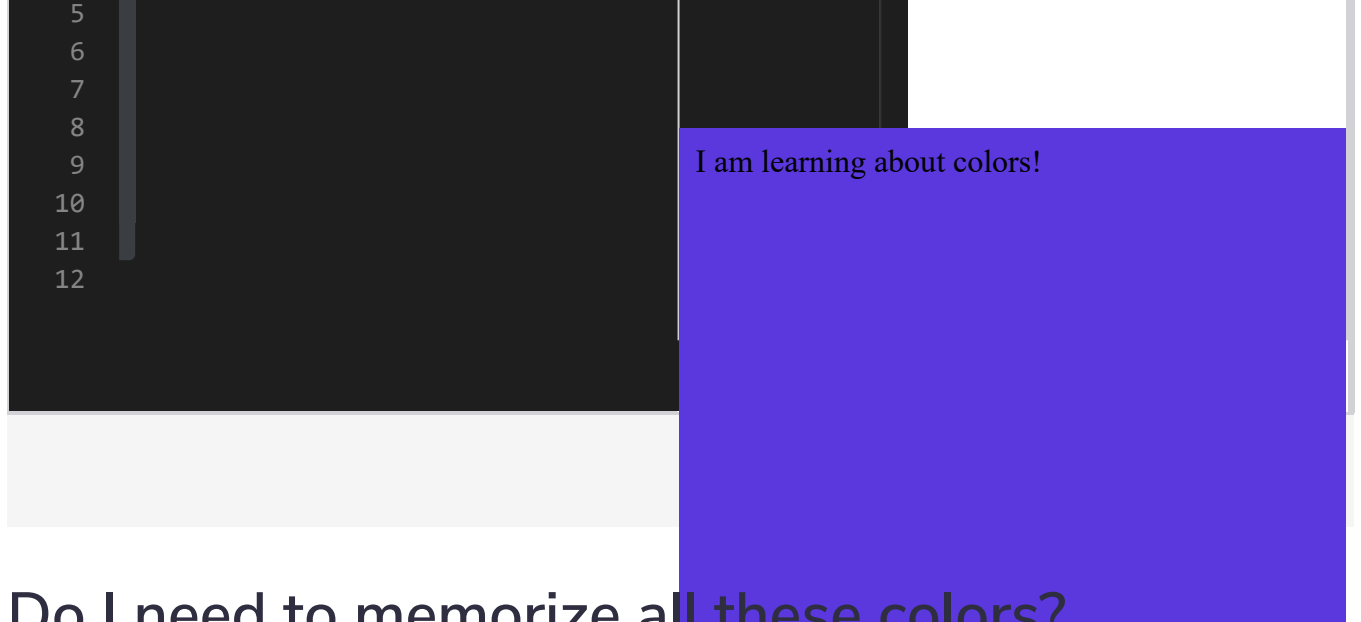
Be sure to toy with the code in the playground below:


```css
1  body {
2    background-color: ▮rgb(90, 56, 222)
3  }
4
```

```
 5
 6
 7
 8
 9
10
11
12
```

I am learning about colors!

Do I need to memorize all these colors?

No you dont. There are many tools that exist to make the process of choosing colors for your web projects a lot easier.

One of such, and perhaps my favorite is https://www.materialui.co/colors

Dont worry, in the practical sections that come, we will build real user interfaces and garnish them with colors. I'll show you all the tricks of the trade 😁

2. RGBA Values

RGBA values are very much like RGB values - except for one thing. The addition of transparency.

Sometimes you don't want a solid color but want to sprinkle a bit of transparency. In such cases, make use of RGBA values.

The "A" after "RGB" in "RGBA" stands represents the color's alpha channel. This specifies the opacity of the object.

Let's see an eample

Example

```
p {
    background-color: rgba(0,255,0,0.8)
}
```

The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (fully opaque).

Thus, the example above will yield a slightly transparent green color.

Curious what this looks like, you can test this in any of the playgrounds above.

2. The HEX Value

Colors in CSS may also be defined using something referred to as HEX values.

They look rather weird. Something like, `#8cacea` . Who invented this? and why is it so confusing?

The good news is, you don't need to concern yourself with the origin of the weird fellow.

Here is what matters. Concern yourself with these:

1. The *"Hex"* in Hex value, refers to Hexademical.
2. The general format is `#RRGGBB` where R is Red, G is Green and B is Blue.
3. Unlike RGB values that span `0 - 255` , hex values span `00 - FF` where `00` is the lowest and `FF` the highest.

A Quick Example

This would give a red color:

```
p {
    background-color: #FF0000
}
```

In like manner, `#00FF00` will give a Green color and `0000FF` , Blue

Fair Warning

Like I said before, please don't overthink colors for now. I will walk you through some color magic during the practical sections. All you need for now is the basics of how these colors are formed - that's all really 👍

Conclusion

And that's it! There's a lot of fun to be caught in the practical sections. Let's move on 😉