

Lecture 04: Color Spaces and JavaScript

COSC 225: Algorithms and Visualization

Spring, 2023

Announcements

1. Assignment 01 Grading (input validation)
2. Assignment 02 Due Tonight
3. Quiz 02 Wednesday 02/15 (CSS basics)
4. Assignment 03 Due Friday, 02/15 (JavaScript)
 - mostly uses elements from today's class

Outline

1. Activity: Flag of Many Colors
2. Color Spaces
3. Introducing JavaScript

Last Time: Representing Colors

A color that can be represented on a computer screen is represented by three values:

1. intensity of red sub-pixel
2. intensity of green sub-pixel
3. intensity of blue sub-pixel

Color is a three-dimensional object!

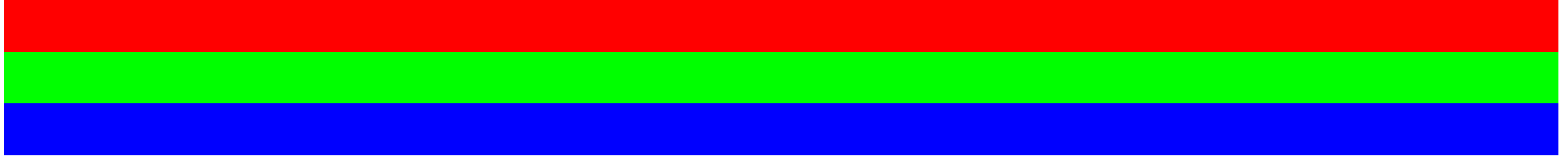
In HTML: `rgb(red, green, blue)`

- red, green, blue are integers from 0 to 255.
- $256^3 \approx 1.7$ million colors!

Observation

Manipulation of r , g , b color values is **not** intuitive

- red, green, blue have natural **physical** interpretations



- *combinations* of red, green, blue do not have natural **perceptual** interpretations (at least to me)



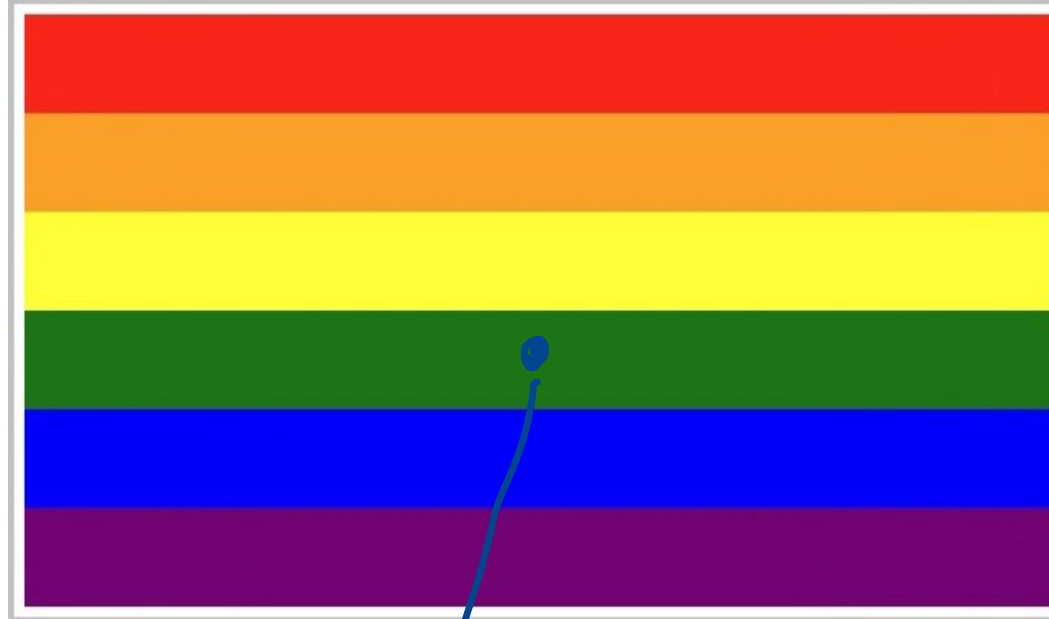
Question. What are the RGB values of the color above?

Let's Make a Rainbow

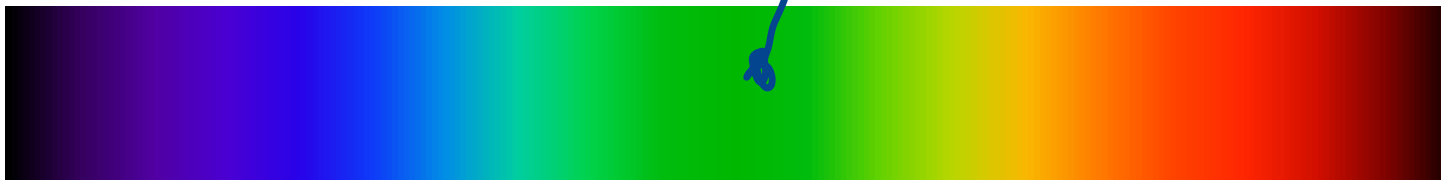
Used predefined colors:

```
<div class="flag">  
  <div style="background-color: red;" class="stripe"></div>  
  <div style="background-color: orange;" class="stripe"></div>  
  <div style="background-color: yellow;" class="stripe"></div>  
  <div style="background-color: green;" class="stripe"></div>  
  <div style="background-color: blue;" class="stripe"></div>  
  <div style="background-color: purple;" class="stripe"></div>  
</div>
```

The Result



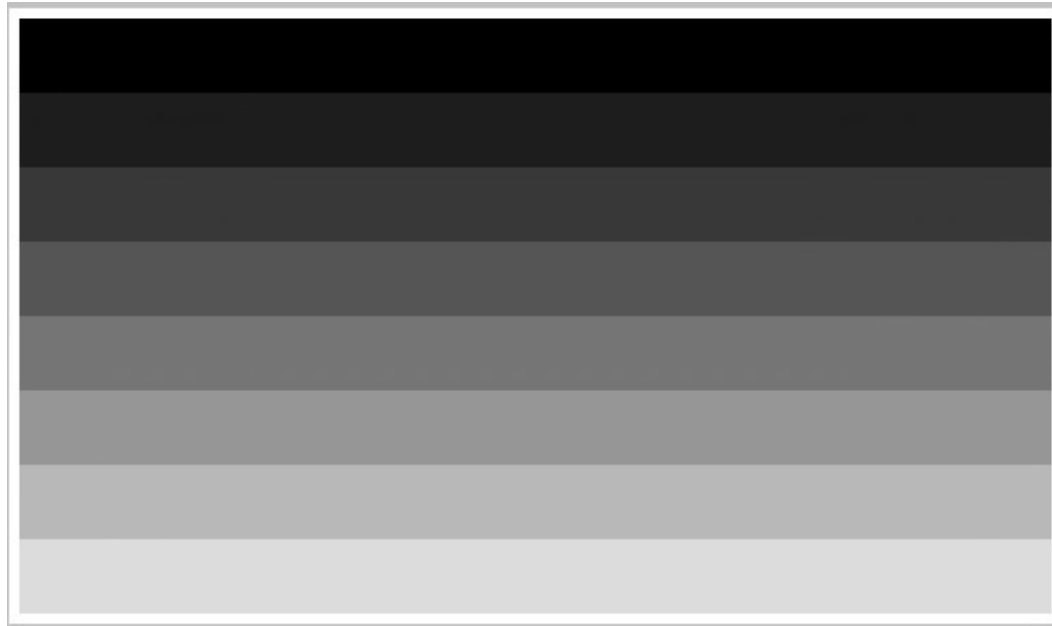
Question. What do you think of HTML's color choices?



Activity (Pairs)

Make a rainbow with 8 stripes!

- use RGB colors
- how to interpolate color values to make rainbow?



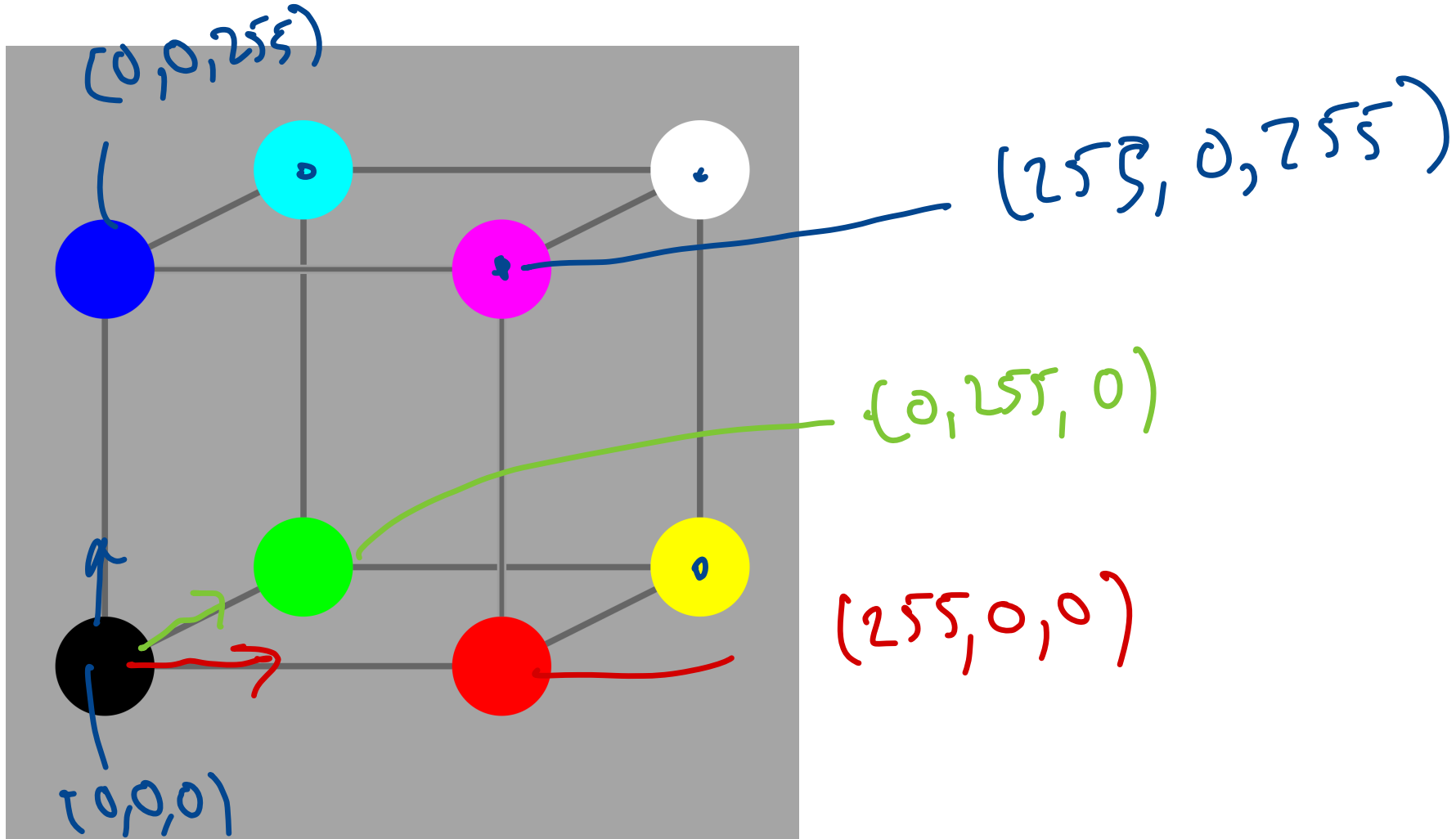
Download [rainbow-eight.html](#) to get started, use RGB color picker

Questions

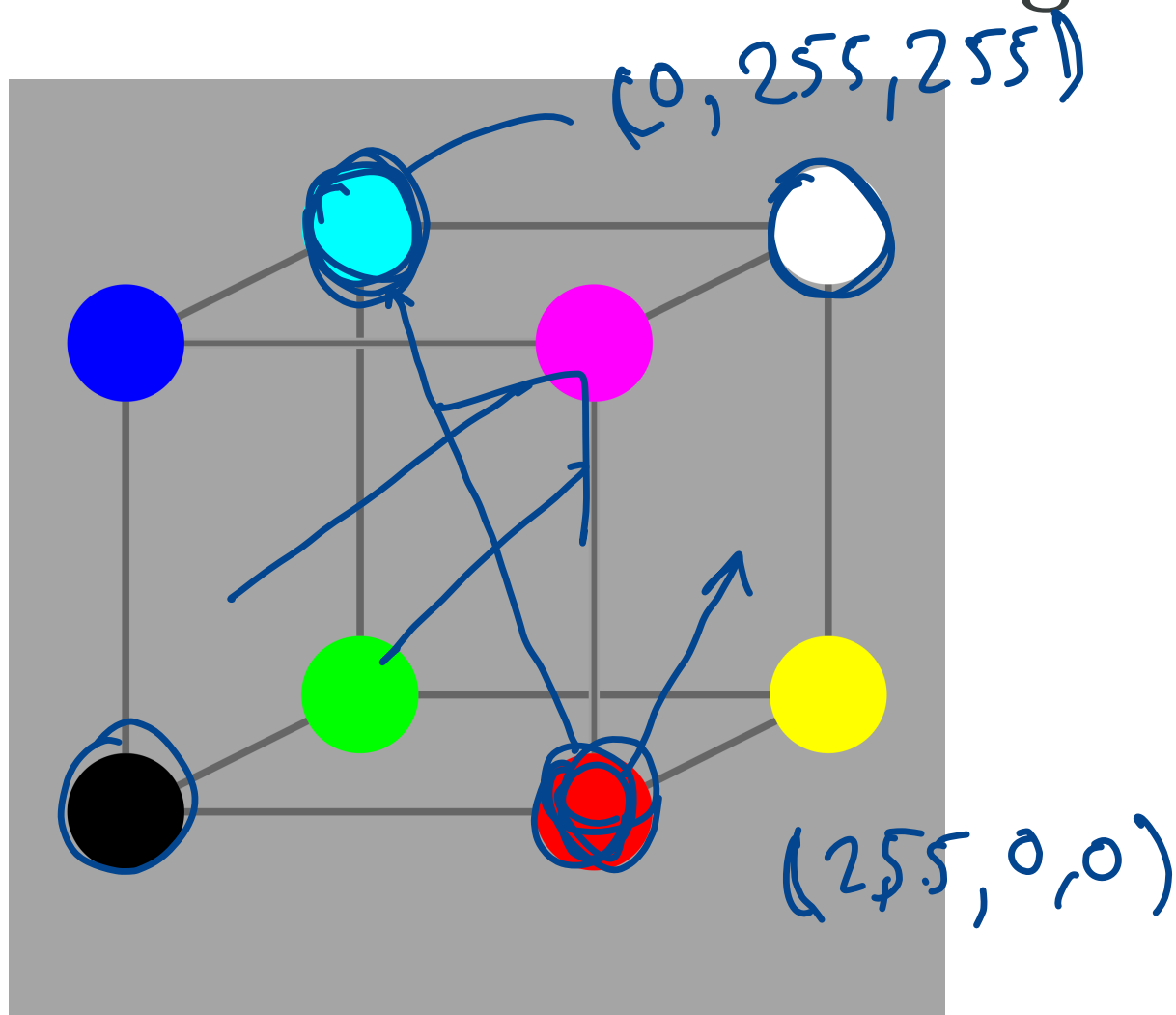
1. What RGB values did you use for the stripes?
2. Is there a pattern of how to pick the color of the next stripe?
3. How do combinations of RGB values relate to your perception of the colors?
 - What adjectives would you use to describe the colors you picked?
4. Do colors look similar on your screen and the projector?

Colors, Geometry, and Perception

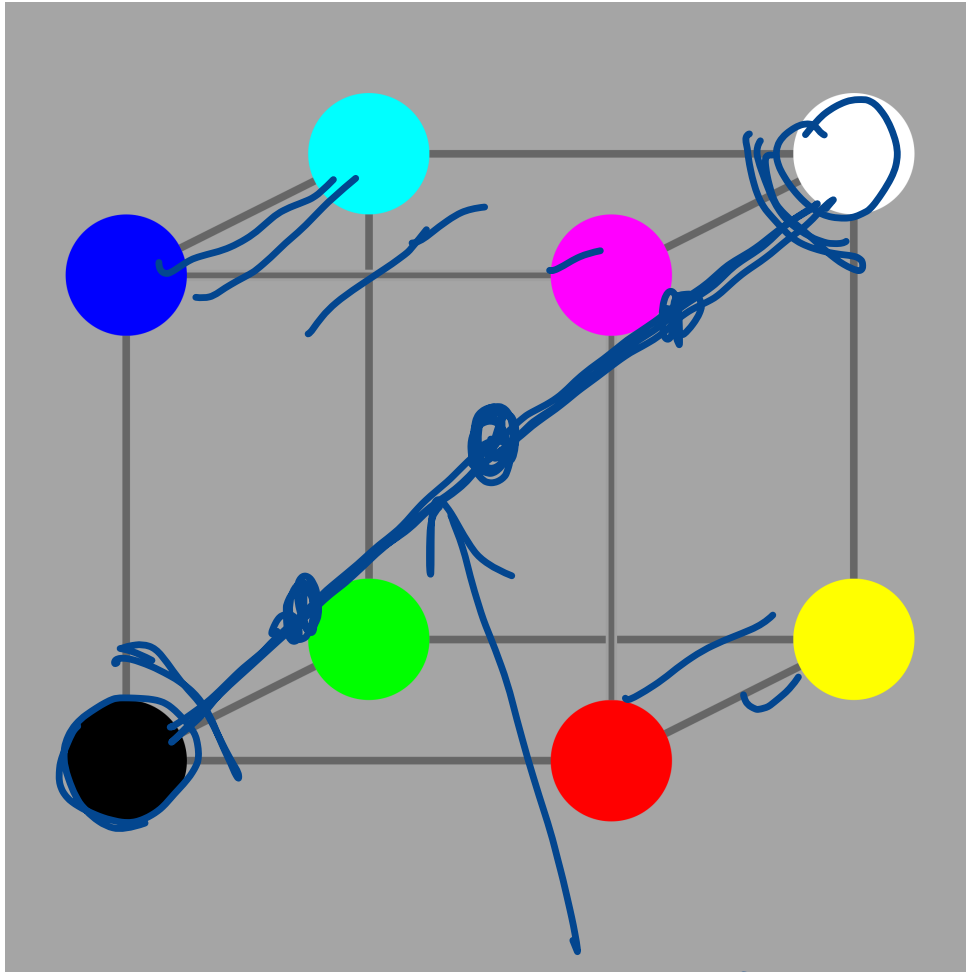
A color is a 3D object: interpret RGB values as coordinates of points in 3D space



Which Colors are “Lighter”?



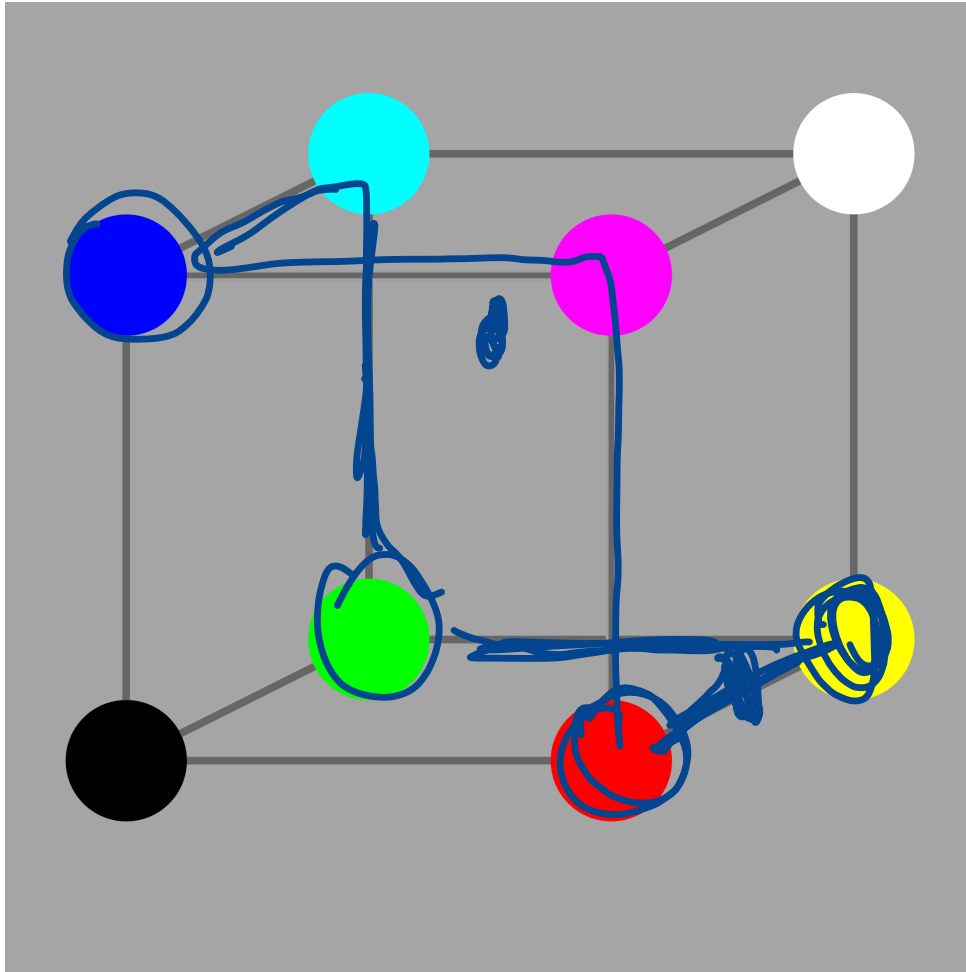
Which Colors are more “Saturated”?



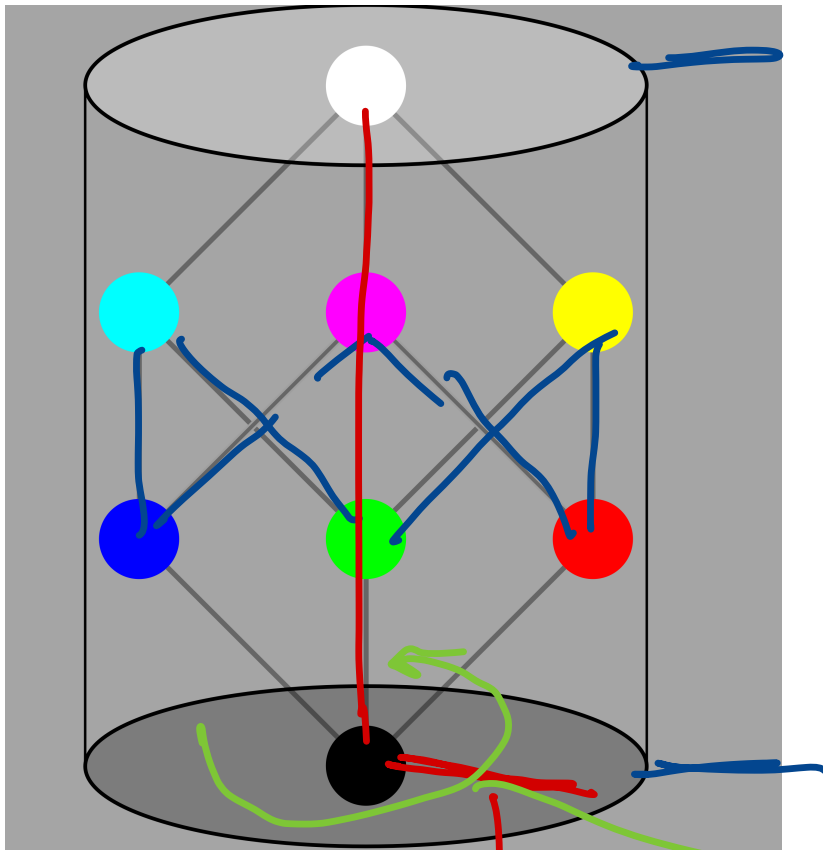
gray lives here

Saturated
=
far from
(gray)
diagonal

Where are “Pure” Hues?



Cylindrical View



↑
lightness
↓

angle reps.
hue

↔ Saturation ↔

RGB vs HSL

Perceptual Dimensions:

- **Hue** the “pure” color as represented on a rainbow
- **Saturation** “intensity” of color
- **Lightness** how light (bright) the color appears

RGB vs HSL

Perceptual Dimensions:

- **Hue** the “pure” color as represented on a rainbow
- **Saturation** “intensity” of color
- **Lightness** how light (bright) the color appears

Can define mathematical relationship between HSL and RGB coordinates

- one-to-one correspondance
 - every RGB point has corresponding HSL value
 - every HSL point has corresponding RGB value
- mathematical relationship between
 - how colors are produced (RGB monitor)
 - how colors are perceived (HSL)

HSL in CSS

RGB:

```
color: rgb(red, green, blue);
```

- red, green, blue are integers from 0 to 255

HSL:

```
color: hsl(hue, saturation, lightness);
```

- hue is a number (degrees), nominally from 0 to 359
- saturation and lightness are percentages (0% to 100%)

HSL Color Picker Demo

Other Color Spaces

There are infinitely many ways to represent colors!

- RGB and HSL are just two

Others made for different hardware/aspects of perception

- RGB and HSL are “additive” color spaces
- subtractive spaces, e.g., for paint/dye mixing
 - CMY(K)

Vision Differences

No all people have all three types of color receptors!

- color blindness affects ~5% of population

Universal design: make graphical that are visually distinctive

- lightness vs hue/saturation
- patterns, not just color

Tool: Firefox color vision simulation

- WebDev Tools -> Accessibility Tab -> Simulate

JavaScript

So Far...

- HTML specifies document content, structure, semantics
- CSS specifies display

And now

- JavaScript specifies **interactions**

So Far...

- HTML specifies document content, structure, semantics
- CSS specifies display

And now

- JavaScript specifies **interactions**

With JavaScript we can

- create/remove elements
- modify elements
- define user interactions

Today

Creating and adding elements to a site!

`hello-javascript.zip`

JavaScript, Two Ways

```
<!doctype html>
<html lang=en>
  <head>
    <meta charset=utf-8>
    <title>Page Title</title>
    <script src="hello.js"></script>
    <script>
      ...javascript code here...
    </script>
  </head>
</html>
```

Take a Look

- `hello.js`

Basic Tasks

- Get an element in the document (selector is like CSS selector) *will always ref same elt*

```
const someElement = document.querySelector("selector");
```

first element in document matching selector is returned

- Create an element (some-tag is desired tag of element)

```
let myElement = document.createElement("some-tag");
```

- Add text to element *can change elt this refers to*

```
myElement.textContent = "some text";
```

- Add element as child of another

```
someElement.appendChild(myElement);
```

Adding Style

If someElement is an element, we can...

- set an id

```
someElement.id = "some-id";
```

- add a class

```
someElement.classList.add("some-class");
```

array of classes assoc. w/
some elt

- add a style

```
someElement.style.backgroundColor = "rgb(200,200,200)";
```

CSS: dashes : background-color
JS : camel case : }

Activity

Let's style our example site!

Next Time

- Visualizing Simple Machines: Cellular Automata!