Lecture 03: Color and Style

COSC 225: Algorithms and Visualization

Spring, 2023

Announcements

1. Assignment 02 Due Monday, 02/13
2. Quiz 02 Wednesday 02/15 (CSS basics)
3. Assignment 03 Due Friday, 02/15 (short)

Outline

1. CSS Crash Course
2. Color

Last Time

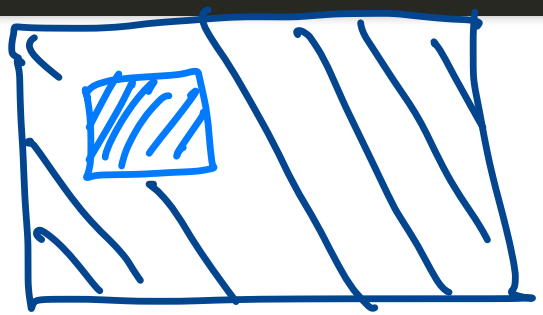
Colored Boxes!

To Draw a Box

document division elt

- use `div` elements
- use `style` attribute to specify
  - dimensions (`width`, `height`)
  - positioning system (`position`: `static`, `relative`, `absolute`)
  - position coordinates (`top`, `left`)
  - color (`background-color`)

```
<div style="background-color: black; width: 300px;
    height: 300px; position: relative;">
  <div style="background-color: blue; width: 100px;
      height: 100px; top: 50px; left: 50px; position:
      absolute;"></div>
</div>
```



Observe

Writing `style = "..."` for every element is cumbersome!

- a lot of text to draw a single box
- making changes is a pain
- many boxes may have similar `style` attributes
- no *semantics* associated with `style` attribute values

Setting `style` manually is not good style!

... if only there was a better way...

Introducing CSS

Cascading Style Sheets (CSS): specify `style` attributes for many elements on a page:

- all elements with same tag (e.g., `h1`, `p`, `div`)
- all elements of same class (set attribute `class="some-class"`)
- an element with a specific id (set attribute `id="some-id"`)

Change to CSS styles affect all elements matching a prescribed pattern

Boxes Revisited

Example from before

- big outer box
- smaller boxes inside

Updating the HTML

Rather than setting `style` attribute directly, specify **semantics**

- use `class` attribute to give names to the *types* of boxes

```
<div class="outer-box">
  <div class="inner-box"></div>
  <div class="inner-box"></div>
</div>
```

Pick Style for Boxes by Type

All boxes (`div` elements):

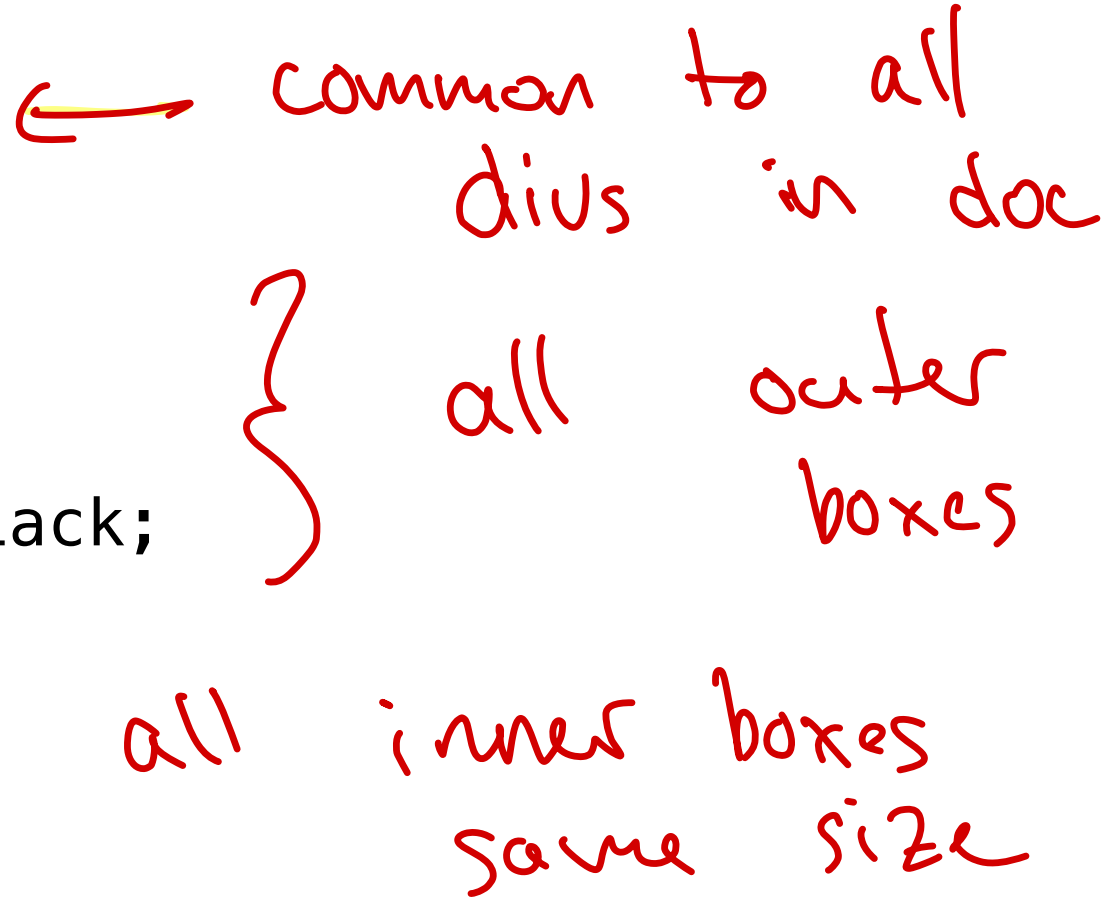
- `position: absolute;`

Outer box:

- `width: 300px;`
- `height: 300px;`
- `background-color: black;`

Inner boxes:

- `width: 100px;`
- `height: 100px;`



Where to Put CSS Styling?

Two methods:

1. In `.html` head:

```
<head>
    <style>
        /* style goes here */
    </style>
</head>
```

2. A separate file, say `style.css`, then add reference in head:

```
<head>
    <link rel="stylesheet" href="style.css">
</head>
```

CSS Style

tag style applies to

- apply style to all `div` elements in the document

```
div {
  position: absolute;
}
```

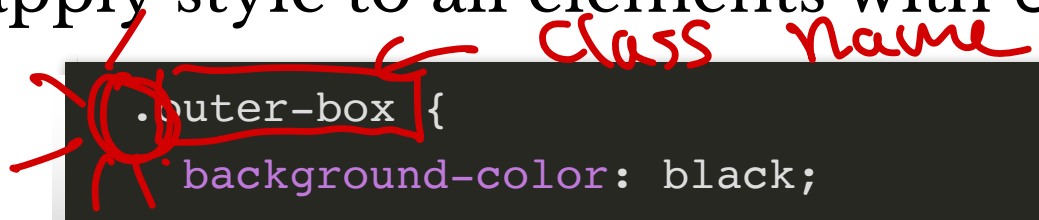
← set style attrib. for all divs

CSS Style

- apply style to all `div` elements in the document

```
div {
  position: absolute;
}
```

- apply style to all elements with `class="outer-box"`




```
.outer-box {
  background-color: black;
  width: 300px;
  height: 300px;
}
```

Dealing with Inner Boxes

- apply style to all elements with `class="inner-box"`

```
.inner-box {
  width: 100px;
  height: 100px;
}
```

Coloring Inner Boxes?

Each inner box has own:

- position (`top`, `left`)
- color (`background-color`)

CSS can style element by id!

Give inner boxes unique ids:

```
<div id="blue-box" class="inner-box"></div>
<div id="red-box" class="inner-box"></div>
```

Style Elements by ID


```
#blue-box {
    background-color: blue;
    top: 50px;
    left: 50px;
}

#red-box {
    background-color: red;
    top: 150px;
    left: 100px;
}
```

Example

`stylish-boxes.html`

Notes

1. CSS properties can conflict
   - more specific rules win
     - `id` beats `class` beats tag
   - if same specificity, last rule in css source wins

```
p { color: red; }
p { color: blue;}
```

```
<p>What color is this text</p>
```

reason color: blue is later in CSS spec.

2. Elements can have multiple `classes`, but `id` should be unique

```
<div id="red-box" class="inner-box special-box"></div>
```

Assignment 02

Go forth and make a stylish website!

Interlude

Color and Perception

What is Color?

Color, Three Ways:

1. subjective **perception** of color
2. physical **production** of color
3. formal **representation** of color

What is Color?

Color, Three Ways:

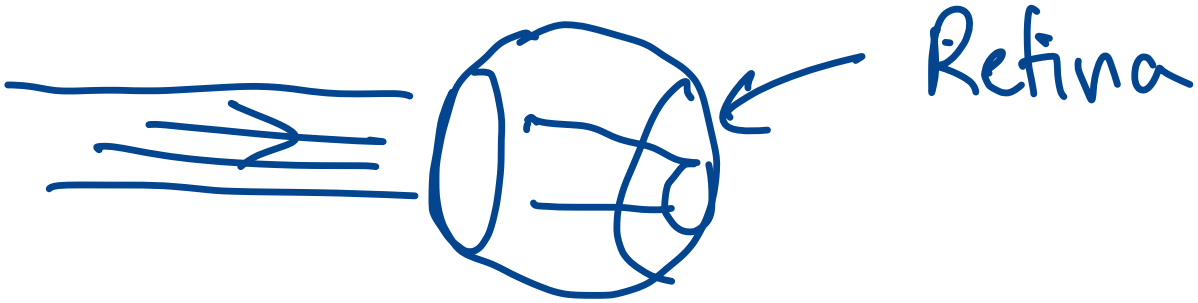
1. subjective **perception** of color
2. physical **production** of color
3. formal **representation** of color

Content Warning: Massive oversimplifications coming up!!!

Color and Light

Physics $\Longrightarrow$ Perception

- Color perception begins with **light**
  - light enters the eye
  - light stimulates receptors in the retina
  - retinal stimulation results in perception (somehow)



Color and Light

Physics $\Longrightarrow$ Perception

- Color perception begins with **light**
  - light enters the eye
  - light stimulates receptors in the retina
  - retinal stimulation results in perception (somehow)
- "Pure" light has two attributes
  - wavelength: hue (e.g., blue, green, red)
  - intensity: brightness

image source: Wikipedia

Perception of Pure Light

- "Trichromatic" humans have three types of color receptors (cones) in their retina
- each receptor has characteristic sensitivity to different wavelengths

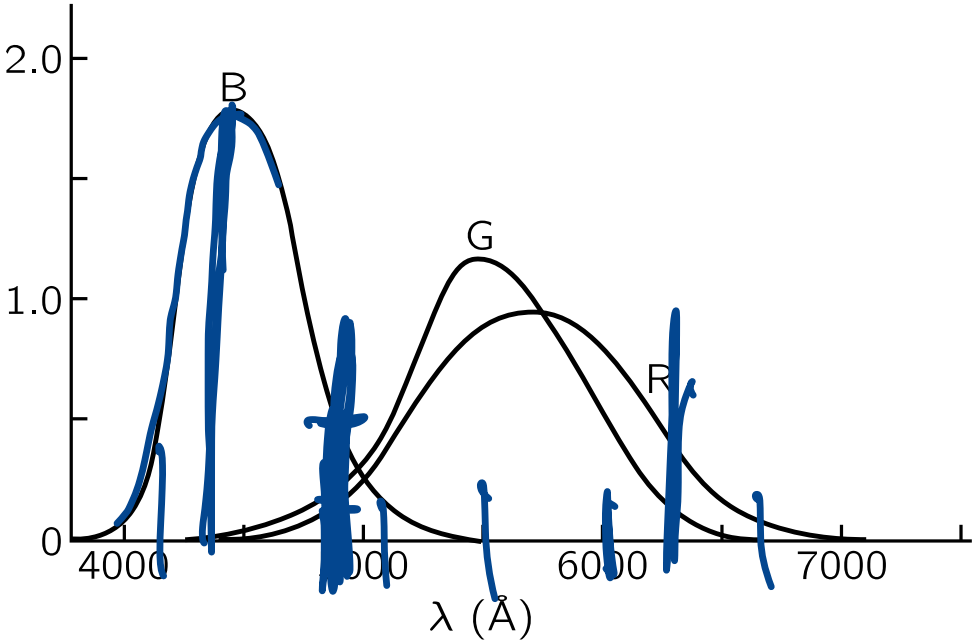


image source: Feynman Lectures on Physics

Natural Light

"Natural" light comprised of different wavelengths in different proportions

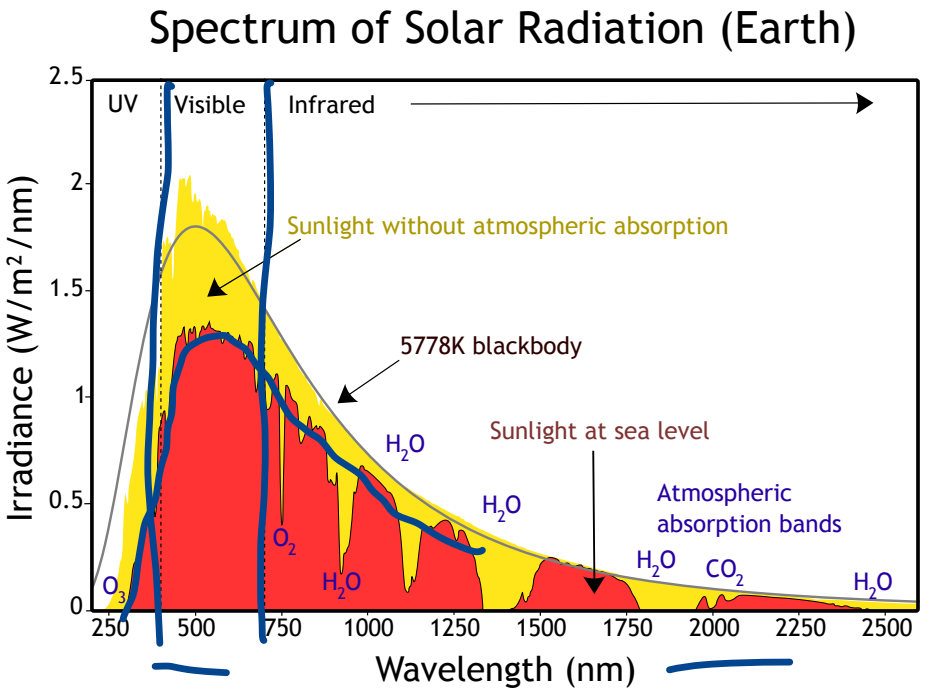
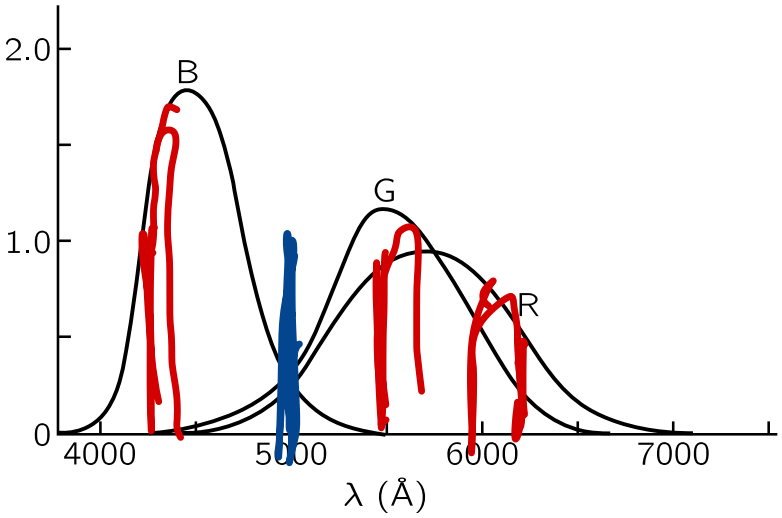


image source: Wikipedia

Perception of Color

Perception of color determined by the amount each color receptor is stimulated

- many different light power spectra correspond indistinguishable colors
- $\Longrightarrow$ it is possible represent many colors by “mixing” a fixed set of colors



Generation of Color

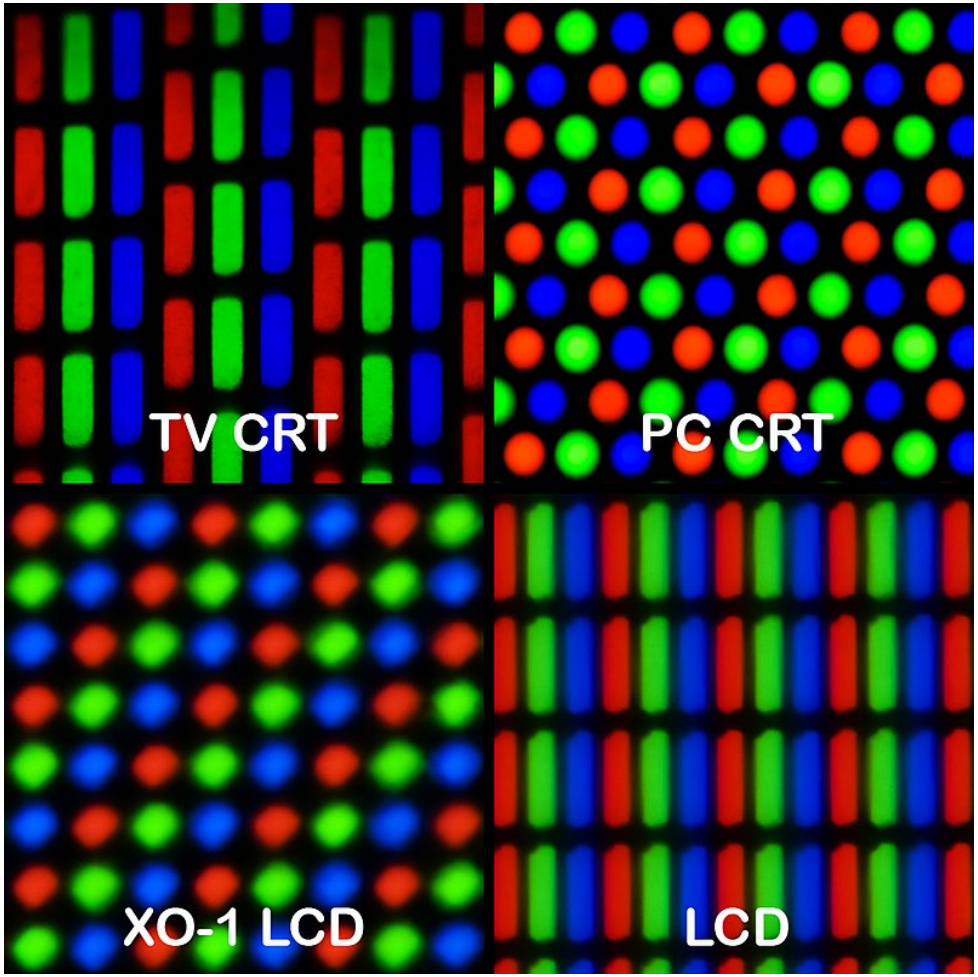
Question. How do color monitors/projectors create so many colors?

Generation of Color

Question. How do color monitors/projectors create so many colors?

- Display is a 2d grid of **pixels**
- Each pixel contains multiple (3?) light producing elements
  - red
  - green
  - blue
- Intensities of each element can be controlled independently

Different Types of Displays



Engineering $\Longrightarrow$ Perception

So far:

- can generate light with different characterstics:
  - vary intensity (brightness) of three different pixel elements
  - red, green, blue
- light emitted by pixels stimulates retina
  - red pixel light stimulates red cones more
  - …
- relative stimulation of different cones in retna $\Longrightarrow$ perception of different colors

Formal Representation of Color

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)`

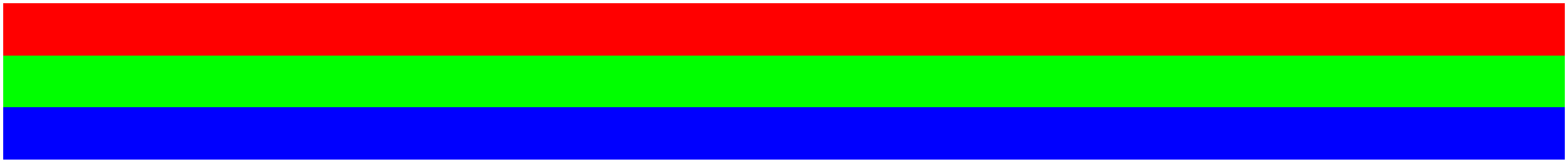
- `red`, `green`, `blue` are integers from 0 to 255
- $256^3 \approx 1.7$ million colors!

Color Picker Demo

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)

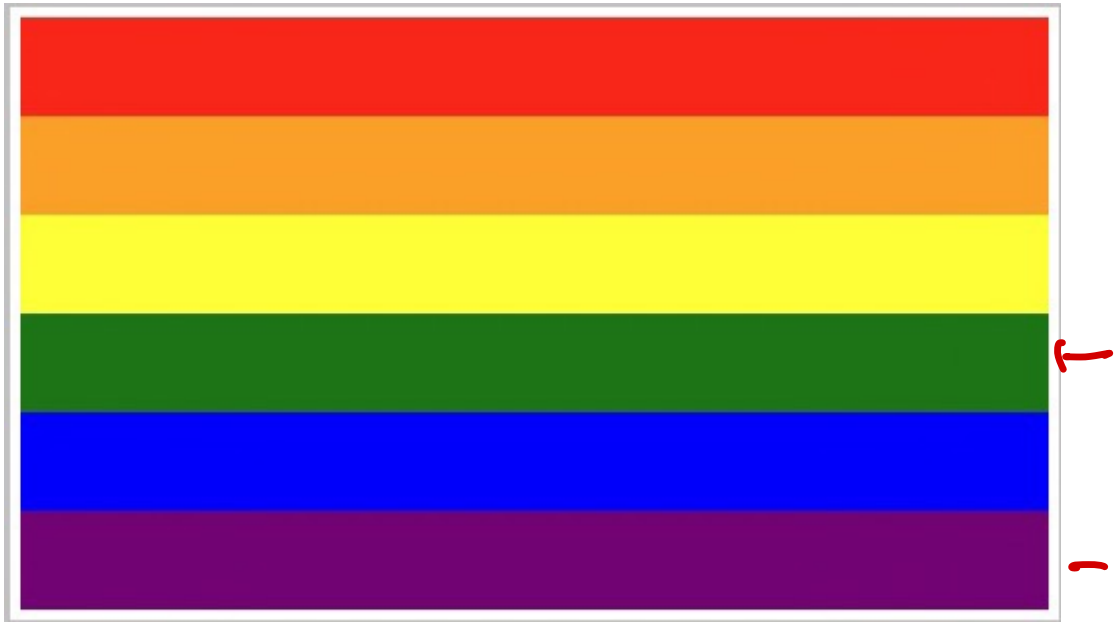
Question. What are the RGB values of the color above?

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>
```

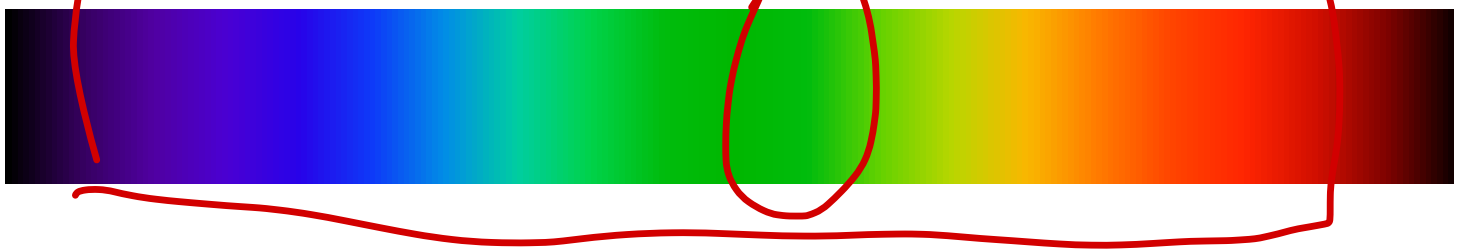
The Result



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

The Result

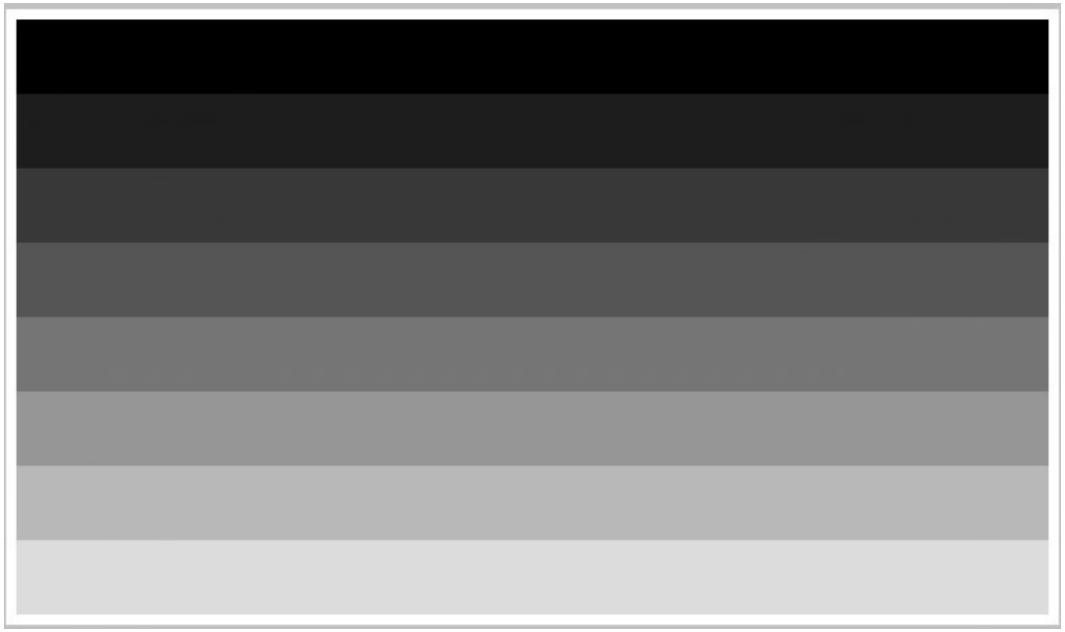
Question. What do you think of HTML's color choices?



Activity (Pairs)

Make a rainbow with 8 stripes!

- use RGB colors
- how to interpolate color values so flag looks “rainbowish?”



Dowload `rainbow-eight.html` to get started

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?

Next Time

1. Colors + Geometry = Color Spaces
2. Introducing JavaScript