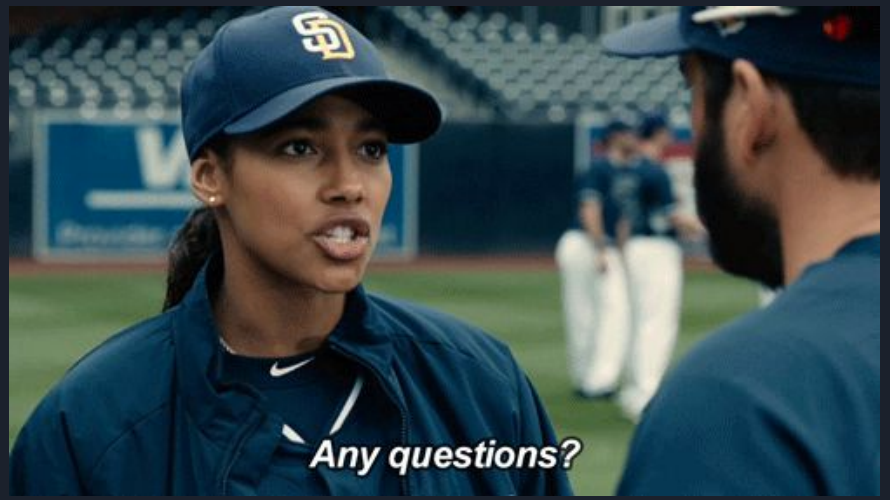
Lecture 3: CSS

CSS: Questions over reading

- CH 7
  - Colors
  - Transparencies
  - URLs
  - Rules
  - Properties
  - etc.
- CH 9
  - Selectors
  - Relationships
  - Pseudo Elements
  - Pseudo Classes
  - etc.



CSS: Last Week Review

- More in depth look into HTML
  - Why HTML is important
  - Browsers are designed to render HTML
  - HTML has 3 main components
    - Text
    - References
    - Markup
  - Parts of HTML Doc
- Set up working ‘class-directory’
- HTML Semantics
- Learned what HTML is meant to do and not do
- Type of Elements
  - Block
  - Inline
- Accessibility
- SEO
- Elements, Attributes, and Values
- Parent > Child relationships
- URLs, linking items
- File naming

CSS: Image Optimization

- Last week we didn't cover Image Optimization
  a. Download a **LARGE** Google Image
  b. Open https://tinypng.com/
  c. Drag & Drop or Select from filesystem
  d. Compare Images
- Do note I have seen issues with this and shadows on .png files with terrible results

Image Optimization in my experience is a great way to cut a ton of weight on your site loading

Submit all your wireframes

steve_fuller@berkeley.edu

Wireframes Needed

- Media Page
- Merch Page
- Tour Dates Page
- Homepage
- About Page
- Discography Page

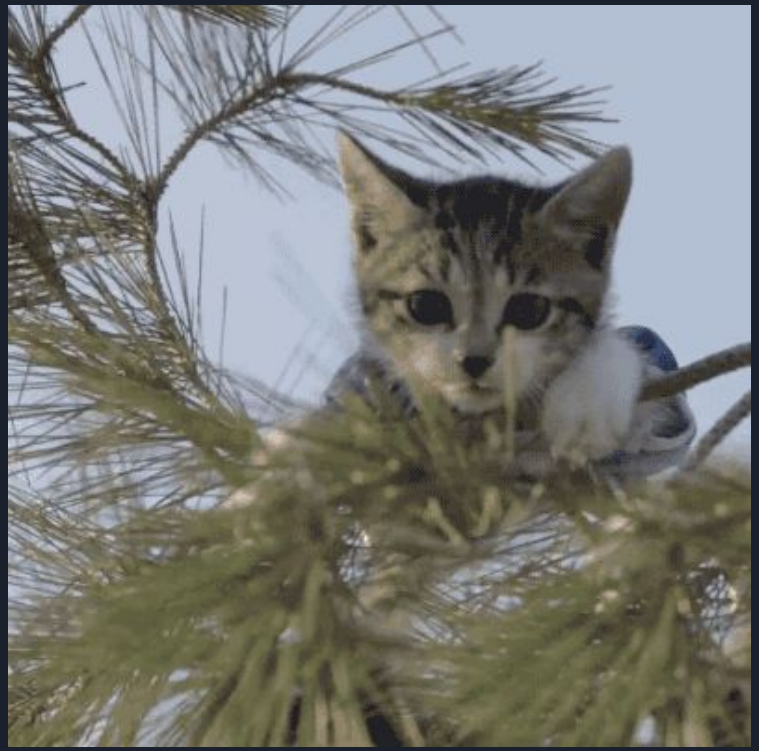
Designing Your Website

Don’t over think this part we are not looking for award winning designs. I want to see what colors you are going with, see some images placed in, and bring some life to your wireframes

- Open Photoshop or Sketch
- Set canvas to 1440px wide
  - Height does not matter when setting up 1024px is a good starting point
- I like to set my working content to 1200px
  - I set guides at 120px and 1,320px
- For ease of designing make spacing of elements by units of 5px
- Grab Custom Fonts from **Google Fonts** **only**
- Bulma UI Kit
- Other Kits
  - Bootstrap 3 UI Kit
  - 14 Bootstrap 4 UI Kits
  - Free UI Kits

CSS Helpful Resources

- StackOverflow - CSS
- W3Schools - CSS
- Can I Use?
- CSS Cheat Sheet



CSS: What is CSS?

- CSS stands for Cascading Style Sheets
- Where HTML defines your content's meaning CSS defines the appearance
- CSS can describe how HTML elements are displayed on
  - Screens
  - Paper
  - Other Media
- External stylesheets are stored in .css files
- CSS1 was released proposed in '94 and released in '96

CSS: Solving a Big Problem

- HTML was never intended for presentation
- HTML was created to **describe the content** of a web page
- HTML 3.2, released in ‘97, allowed for attributes to affect a web pages presentation
- With CSS1 being released it was able to create a “separation of concerns” between presentation and content meaning

CSS: Constructing a Style Rule

- *Selector* - identifies the element(s) you want to format
- *Property* - a property is always associated with a *Value*
- *Value* - is the allowable option(s) for a *Property*

p { color: red; }

- p - is the *selector*
- color - is the *property*
- red - is the *value*

- Selectors
  - *
    - all
  - Element
    - p,ul,li
  - Class
    - .
  - ID
    - #

CSS: Loading in Styles

- CSS can be brought in with various methods
  - External Files
  - Embedded <style> blocks
  - Inline Styles

1. Open /class-directory
2. Within /assets/stylesheets
   a. New File > styles.css
   b. Add a Comment: /* CSS Comment */
3. Create css-styles.html
4. Add <style> tag in the head
5. Load in styles.css

```
<!doctype>

<html>

<head>

<title></title>

<link rel=”stylesheet” type=”text/css” href=”./assets/stylesheets/styles.css” />

<style></style>

</head>

<body><div class=”myBox”>...</div></body>

</html>
```

CSS: Specificity

“Apart from Floats, the CSS Specificity is one of the most difficult concepts to grasp in Cascading Stylesheets” - Smashing Magazine

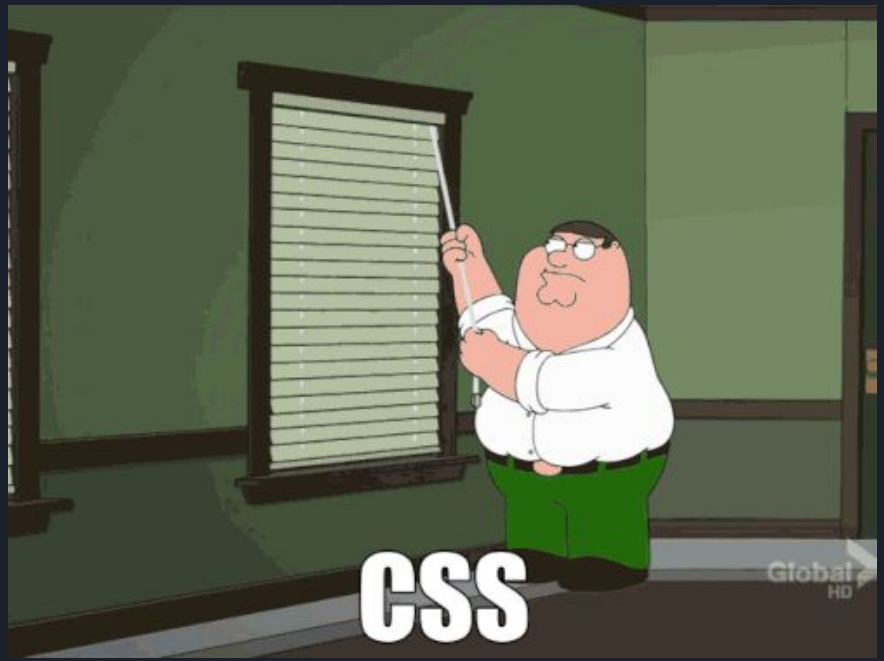
- Determine which CSS rule is applied by the browser
- Every selector has place in specificity hierarchy

5 Categories of specificity

1. Inline Styles
2. IDs
3. Classes
4. Attributes
5. Elements

CSS: Specificity

- Overriding styles can be done in a couple of ways
- When overriding styles things can become confusing very quickly so do this with caution



CSS: Specificity examples

CSS: Inheritance

Inheritance and the cascade are two fundamental concepts in CSS.

- Inheritance is associated with how the elements in the HTML markup inherit properties from their parent elements and pass them on to their children
- The cascade relates to CSS declarations being applied to a document, and how conflicting rules do or do not override each other

This is important because it keeps your code DRY (Don’t Repeat Yourself) by cutting down on redundant code.

CSS: Inheritance Example

CSS: Block Elements

- Inline Elements
  - Respect Left & Right margins and padding but **not** top & bottom
  - Cannot have a width and height set
  - Allow other elements to sit to their left and right
- Block Elements
  - Respect all
  - Force a line break after the block element
- Inline-block Elements
  - Allow other elements to sit to their left and right
  - Respect top & bottom margins and padding
  - Respect height and width

Background: the difference between div and span

W3schools Block-level Elements

CSS: Block Example

CSS: Pseudo Classes

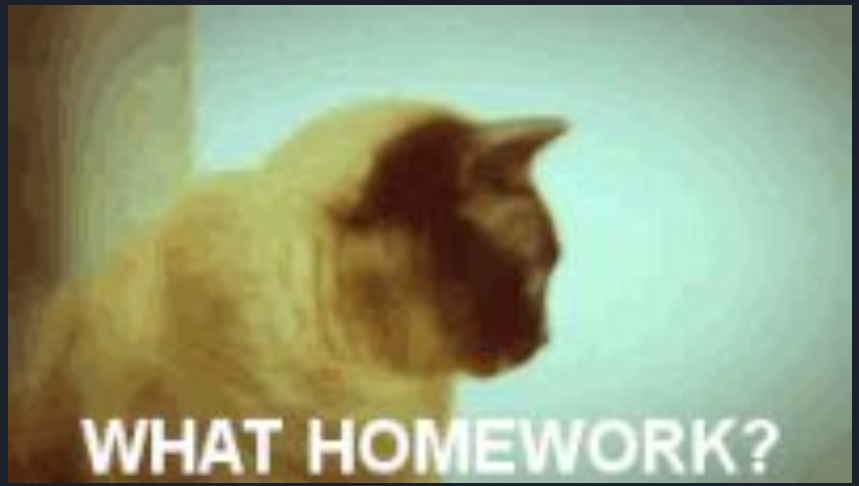
- Defines a special state of an element
- Style an element when you perform an action

selector: pseudo-class { property: value; }

MDN web docs Pseudo-classes

Pseudo Class Examples

Week 3: Homework



- Start Designing your Website
  - Finish Desktop Designs for:
    - Homepage
    - About
    - Discography
- Read Chapters
  - 10 & 11