

A decorative graphic on the left side of the slide. It consists of a blue parallelogram and a light green parallelogram, both tilted at an angle. The blue shape is in the foreground, and the green shape is partially behind it. They are set against a dark blue background with subtle diagonal lines.

# Lecture 3: HTML Deep Dive



# HTML DD: Last Week Review

- What is HTML
- History of HTML
- Introduction to Atom Text Editor
- Hello World Page
- HTML Tags
- Semantic HTML

# HTML DD: Questions over reading

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





# HTML DD: This week overview

- Design Your Site
- Semantic HTML
- Developer Tools
- Class Directory
  - What it is
  - Setting up
- Naming files
- URLs, Linking, Images
- Image Optimization
- Some CSS

# Designing Your Site





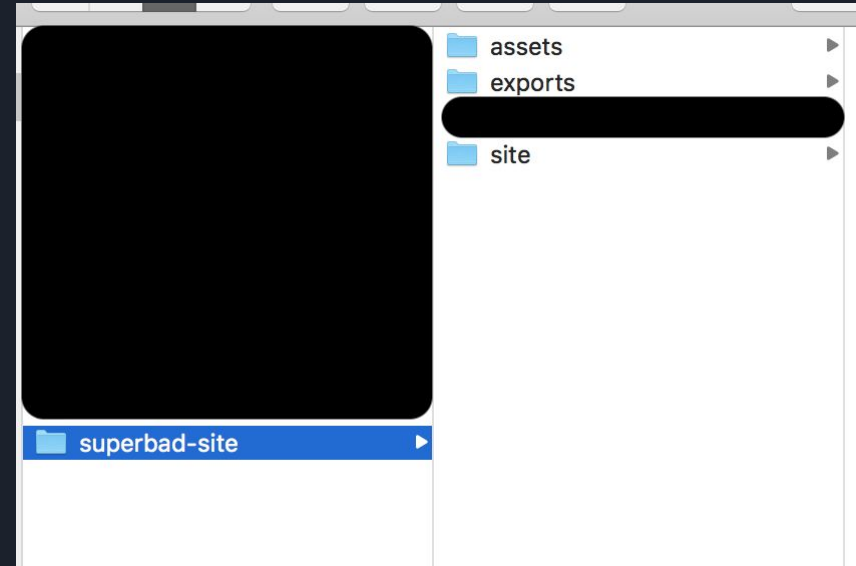
# Designing Your Site

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](#)

# Designing Your Site

- Set up a “Branding” canvas
  - Font sizes
  - Color Palette
  - Font Family(ies)
    - Make sure to [pick a Google Font](#) for easier installation
- File Structure
  - Assets - images and any other external files to design your site
  - Exports - folder to save your files to be loaded into inVision
  - Site - project files



# Designing Your Site



[VIDEO LINK](#)



# Designing Your Site



[VIDEO LINK](#)

# Designing Your Site



[VIDEO LINK](#)

Let's talk HTML





# HTML DD: Nesting Elements

- Parents and Children - an HTML element that contains another HTML element is considered a “Parent Child relationship”
  - Ex. `<article><p>...some text...</p></article>` | The `<article>` is the *Parent* to the *Child* `<p>`
  - Parents can have theoretically infinite nested children although this isn't advised. “nested children” `<article><p>...some text...<ul><li>...</li></ul></p></article>`
- Parent Child relationships are extremely important to understand when you begin working with [CSS](#) and [JavaScript](#)



# HTML DD: Why Semantic HTML matters

- Accessibility
  - Users with some type of impairment usually visual
  - HTML is meant to be universal and all users should be able to view and digest content
  - [YouTube: Introduction to Web Accessibility and W3C Standards](#)
  - [YouTube: Why do semantics matter?](#)
- Improved SEO
  - Search Engine Optimization
    - Search engines look for particular tags on pages and these will be used as part of your site rating
    - [Beginner Guide to SEO](#)
    - Google Analytics Demo
- Styling and maintenance
  - If we are coding to the same standard it makes for a better experience picking up new code more efficiently
  - Makes writing CSS easier and developer understanding



# HTML DD: Semantics contd.

- HTML was a few years old when CSS1 arrived in 1996. Because of this HTML had to handle some presentation qualities
  - Lots of text styles for example
    - Bold
    - Italic
    - Sizing
- HTML presentation fell out of favor with HTML 4 instead pushed towards CSS styling
- HTML5 goes further to make elements less presentational and more semantic
- In the end choose tags that describe the content regardless of presentational effects

KITTENS!!!!

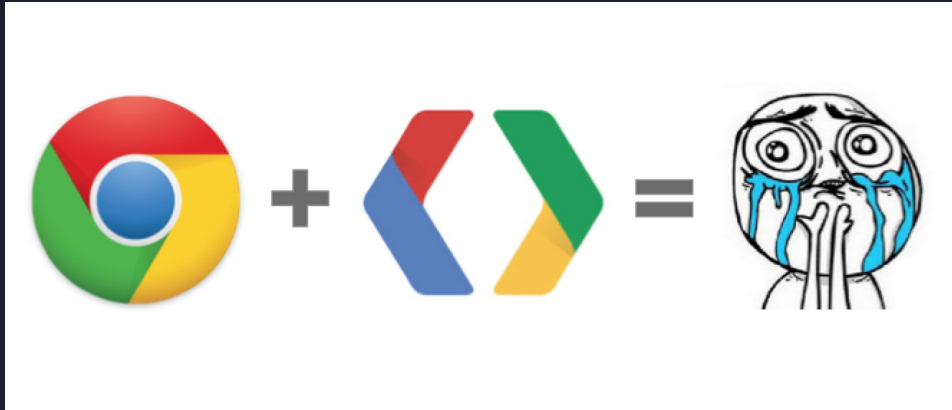


# Chrome Developer Tools





# HTML DD: Chrome Developer Tools



The most invaluable source for help is the [Chrome Developer Tools](#).

- View and change CSS
- Debug JavaScript
- Simulate mobile devices
- Test responsiveness
- Edit DOM elements
- Inspect animations
- Etc.

Needless to say developer tools is the more crucial way to make sure your site or application is performing as expected

# HTML DD: Accessing Chrome Developer Tools



## Chrome Developer tools

In creating websites and web applications one of the most important tools is the developer tools.

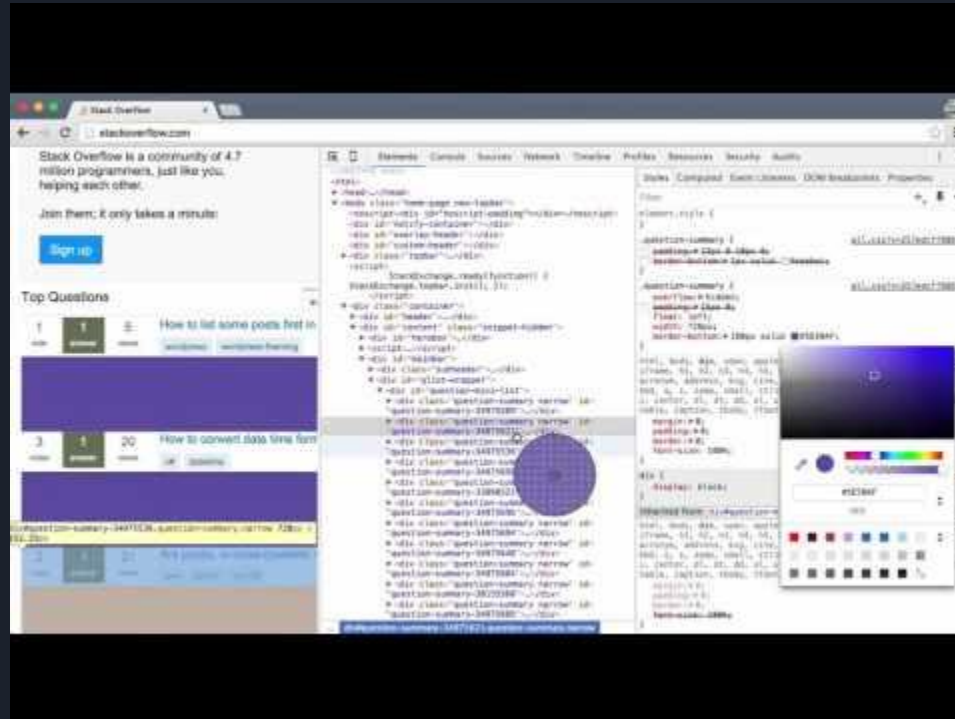
Every modern web browser includes a powerful suite of developer tools. These tools do a range of things, from inspecting currently-loaded HTML, CSS and JavaScript to showing which assets the page has requested and how long they took to load.

[What are developer tools?](#)

How do you pull it up? Three ways:

- *Keyboard.* *Ctrl + Shift + I*, except
  - *Internet Explorer.* *F12*
  - *Mac OS X.* *⌘ + ⌥ + I*
- *Menu bar.*
  - *Firefox.* Menu   *Toggle Tools*, or *Tools* *► Web Developer* *► Toggle Tools*
  - *Chrome.* *View* *► Developer* *► Developer Tools*
  - *Safari.* *Develop* *► Show Web Inspector*. If you can't see the *Develop* menu, go to *Safari* *► Preferences* *► Advanced*, and check the *Show Develop menu in menu bar* checkbox.
  - *Opera.* *Developer* *► Web Inspector*
- *Context menu.* Press-and-hold/right-click an item on a webpage (Ctrl-click on the Mac), and choose *Inspect Element* from the context menu that appears. (*An added bonus:* this method straight-away highlights the code of the element you right-clicked.)

# HTML DD: Chrome Dev Tools Tutorial



[VIDEO LINK](#)

Let's try the Chrome  
Dev Tools



# Class Directory



# HTML DD: What is the Class Directory?

The Class Directory is meant to serve as a single location for everything we cover in class. It will be the reference you can use to build out your final project.



# HTML DD: Setting up Class Directory

- Select the storage option
  - Laptop
  - USB Drive
  - Cloud Storage

```
/class-directory  
/assets  
  /images  
  .gitkeep  
/stylesheets  
  styles.css  
/javascripts  
  .gitkeep  
/fonts  
  .gitkeep  
index.html
```




# HTML DD: File Names

- **Use Lowercase File Names**
  - media.html, discography.html, photos.html, etc.
- **Separate words with a dash '-'**
  - tour-dates.html, my-profile.html, super-cool-page.html
  - Dashes are preferred by Search Engines
  - Underscores '\_' are not wrong just not recommended
- **Be careful of the extensions you use**
  - proper : .html
  - improper: .htm
  - What you declare is what will render
    - .txt will render a text document and so on







# HTML DD: Links, Images, and non-text content

- Since everything in an HTML document is text anything that isn't text is merely a *reference*
- The reference is nothing more than text
- Browsers are able to interpret all types of files but depending on the browser it may not support particular files
  - Browsers will do their best in finding a supporting application to assist in supporting unsupported files
  - You as a developer can also assist in helping browser find applications and extensions to allow the browser to support a non-native file type



# HTML DD: URLs

- Uniform Resource Locator (URL) is a name for addresses.
  - Contains information about where the file is
  - Parts of a URL
    - `https://www.williamsfuller.com/projects/deftones-site/index.html`
    - Scheme
      - Not Secure: `http://`
      - Secure: `https://`
      - Others: `ftp`, `sftp`, etc.
    - Server Name
      - `www.williamsfuller.com`
    - Path
      - `/projects/deftones-site/`
    - File Name
      - `index.html`



# HTML: URLs contd.

- There are 2 types of URLs
  - Absolute - shows the entire path
    - <https://www.williamsfuller.com/projects/deftones-site/index.html>
    - <https://www.williamsfuller.com/projects/deftones-site/assets/stylesheets/styles.css>
  - Relative - details where a file is relative to where you are located
    - ./ before writing file path will denote “start from this file and look for the reference”
    - If we were on the index.html page and wanted to reference the *styles.css*
      - ./assets/stylesheets/styles.css
- When working within files you will typically just use relative paths to grab all your reference images, stylesheets, scripts, etc. from within your project. If you are referencing files outside of your project on the web you’d use absolute paths.



# HTML DD: 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

# HTML: Week 3 Homework

- Homework
  - Read: Ch 10 & 11
  - Designs
    - Home
    - About
    - Discography
  - Practice Coding
    - Tables
    - Import Images
    - Create lists
    - Create links
    - Import and link up Bulma

