Lecture 2: HTML

HTML: Questions over reading

- CH 3 - Basic HTML Structure
  - HTML Structure?
  - Parts of a document
    - <header>, <footer>, <aside>?
  - Naming elements
    - Classes
- CH 6 - Links
  - Block Level Links
    - Wrapping elements in anchor tag
  - Target Attribute

HTML: Last Week Review

- Hyper Text Markup Language
  - Think of as a Skeleton
  - Holds everything together
- Cascading Style Sheets
  - Think of as Clothes
  - Styling of website
- JavaScript
  - Think of as Muscles
  - Creates functionality

- <!DOCTYPE html>
  - Declares file
- <html>
  - Begins the HTML content
- <head>
  - Instructional information for page
    - Meta tags
    - <title>
    - External CSS
- <body>
  - Content user sees

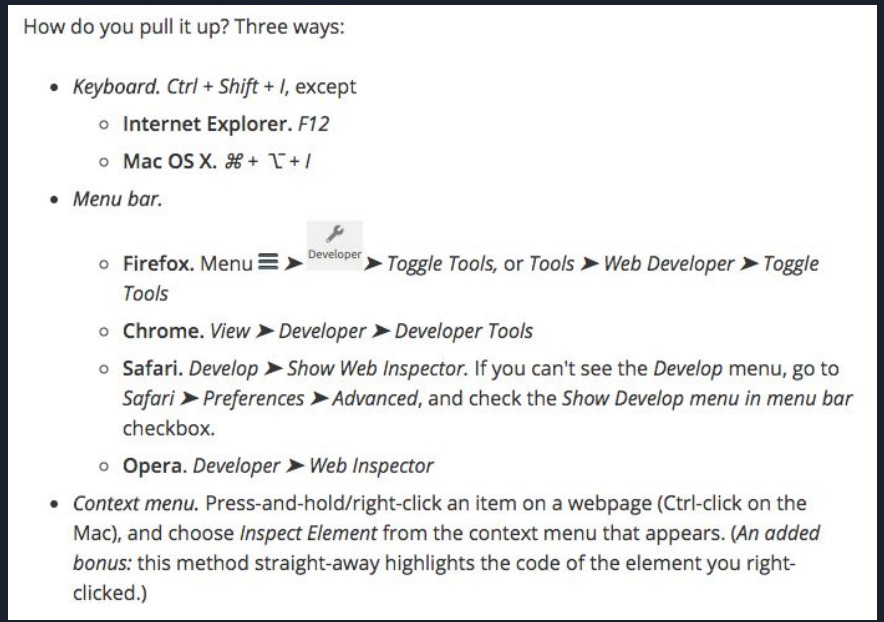
HTML: Last Week Review contd.

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 ☰ ➤ Developer ➤ *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: Goals

Our goals

- Verify wireframes for
  - Homepage
  - About
  - Discography
- Verify band selection
- *Setup personal computers*
- Setup Class Directory on storage device
  - Sign up for
    - Box
    - Google Drive
    - Dropbox
  - *USB Drive or Laptop*

- Understand HTML
  - Semantics
  - Directory Structure
  - Images, Text, Links
  - URLs
  - Element Relationships
- SEO
- Accessibility

HTML: Help Resources

- StackOverflow - HTML
- W3Schools
- HTML Cheatsheet

HTML: Web Pages

- It's impossible to create a web page without HTML
  - New front end frameworks like Angular and ReactJS require HTML but in different ways
    - Angular Example
    - React Example
    - Ruby on Rails Example
- Web Browsers are designed to render HTML for users
- A web page consists of 3 primary components
  - Text Content - bare text that tells the user what your page or site is about
  - References to other files - items such as images, audio, video files or other HTML files
    - Assets - JavaScripts or CSS files
  - Markup - HTML Elements
- These files are all saved as text so they can be universally read by browsers
- There is other information contained in the files not viewable by users
  - Content in the <head>...</head>
  - The <head> contains character encoding usually UTF-8 and other information for the browser and search engines

HTML Exercise: 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: Code Structure

- Let’s create our index.html file basic
- Once complete run atom-live-server

```
<!DOCTYPE html>

<html lang=“en”>

<head>

<meta charset=“utf-8” />

<title></title>

</head>

<body></body>

</html>
```

HTML: Semantics

- HTML is a text document containing “markup”.
- Markup describes the meaning
- The meaning is the Semantics
- HTML doesn’t define how things look, that’s for CSS.
- Every browser has some default styling

- Elements
  - Before HTML5 there were 2 types of presentation elements
    - Block - has its’ own line
    - Inline - displays in a line of text
- HTML5 does away with these concepts because presentation is for CSS
  - Block elements fall into various new HTML definitions based on their semantics
  - Inline is now “phasing content”

HTML: 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 prestación al effects

HTML: Write Semantic Code

HTML: Why Semantics 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: Markup Elements, Attributes, and Values

HTML has 3 principal components: *elements, attributes, and values*

- Elements - labels that describe parts of a page
  - <p>, <h#>, <a>, , <img />, etc.
  - Elements must have a start and stop, however <img /> for example is an “Empty Element” tag meaning it closes without having to declare </img>.
    - Yes:
    - No: <img src=”...” alt=”...”></img>
- Attributes and Values - attributes contain information about the content in the document, not the content itself
  - Google
    - *href* is the attribute | “google.com” is the *value* of *href*
  - You can have one or more attributes.
  - Some attributes have predefined values

HTML: Parent > Child

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

HTML: 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: 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

HTML: 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
    - 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 Exercises

HTML: Review

In depth look into HTML

Covered

- Tags
- URLs
- Accessibility
- SEO
- Semantic code
- Got environments setup
- Element relationships
- File name

Week 3

- Homework
  - Read: Ch 7 & 8
  - Wireframes
    - Media
    - Merch
    - Tour Dates
  - Create final project directory and link to all of the pages