

Going Live

Objectives

In today's class we will...

- Recap the basic topics covered to date
- Work with CSS properties used for adjusting a website's typography
- Utilize custom fonts provided from Google Fonts
- Work with CSS pseudo classes to style elements based on special states
- Utilize the Bootstrap CSS framework for website development

Check-Up Session

How's it going?

After 1 week of Bootcamp, how are you holding up?

What feedback do you have so far?

Instructor Feedback



Seriously, mind-blown.

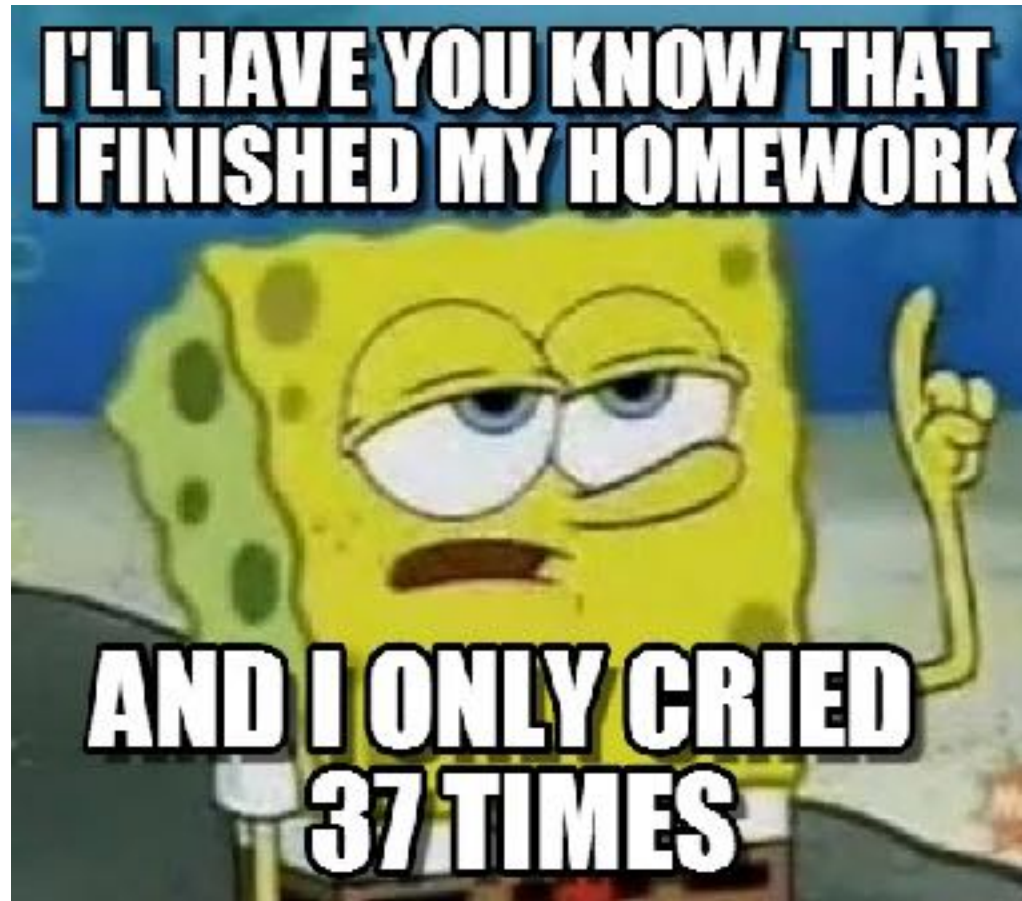
Instructor Feedback

Things I've noticed people doing *incredibly* well:

- All of you are handling an enormous volume of information.
- All of you are asking the right questions.
- You notice the right details.
- You all help each other out.
- And, most importantly, you are **figuring out things on your own.**

A Few Admin Things...

Homework is Due!



Don't forget to submit both the GitHub and GitHub Pages links!

Instructor Feedback

- **Homework Link:**

<<<<<INSERT LINK HERE>>>>>

Remember to submit Homework via GitHub (& GitHub Pages):

<<<<<INSERT LINK HERE>>>>>

And seriously!

Submit whatever you have! Don't get a 0.

(Even if you don't like what you've made.)

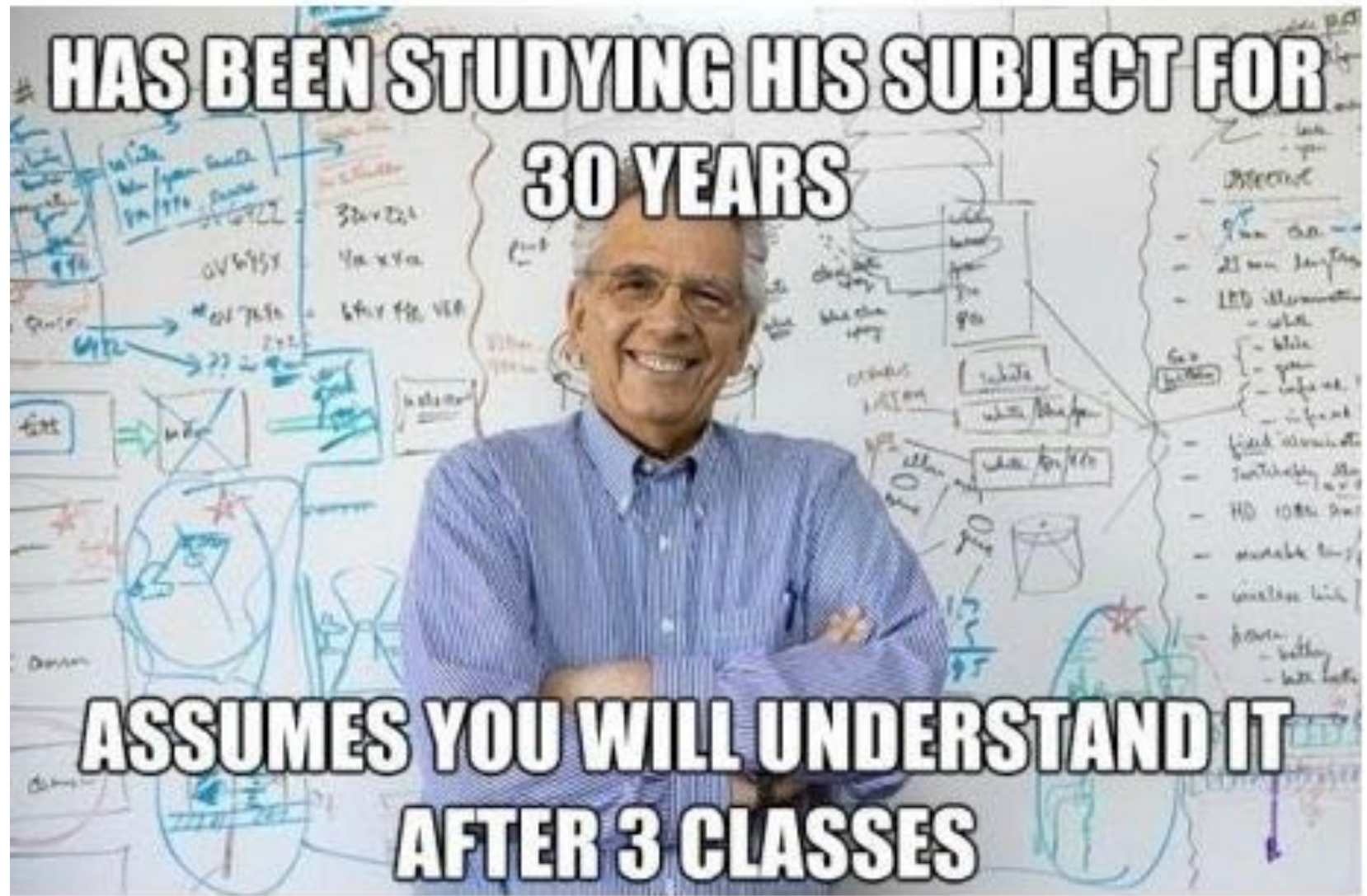
Office Hours + Additional Help

Also, remember....

- **In Class Office Hours: 30** minutes before class, **30** minutes after.
- **Review In Class Material (Exercises and Slides):**
<<<<INSERT LINK>>>

Re-Watch Class Videos:
<<<INSERT LINK>>>>

Feedback #1 – Pace is Fast!!!



Feedback #1 – Pace is Fast!!!

- Here's the reality, the pace of this program **is FAST**.
- We could slow down and make it easier – but that wouldn't prepare you to be developers right out of the program.
- Take up the challenge. Try to retain as much as you can.
- Then on your own time (after the program), go back and review the concepts you felt you needed a refresher on.
- Don't try to master *every* little detail.
- **Focus on the big picture and how to find what you need, when you need it.**

Feedback #1 – Pace is Fast!!!

- That said, as instructors and TAs, we are here to help.
- Share your GitHub code as you run into issues. We can let you know if you're going in the right (or wrong) direction.
- Come to office hours! We can help go over old code, offer tips on homework, or just generally give you a pep talk.
- Reach out to Student Success if you'd ever like to schedule a 1-on-1 meeting.



Feedback #2 – Books Please?



Feedback #2 – Books Please?

- **Advice:**

While in this program, concentrate all your efforts on what we're covering in class.

- Books can be helpful when you are learning on your own. But in a Bootcamp like this, they can also be distracting—especially when our class is moving so quickly.
- When you're at home, focus your efforts on completing your homework, re-completing in-class exercises, and going over slides.
- Invest in books and outside tutorials *after* you finish the program.

Feedback #2 – Books Please?

*That said,
if you truly feel the need for outside resources...*

Feedback #2 – Books Please?



- **Code School** offers some of the best online courses for HTML, CSS, JavaScript, Node, Express and Angular.

Feedback #2 – Books Please?



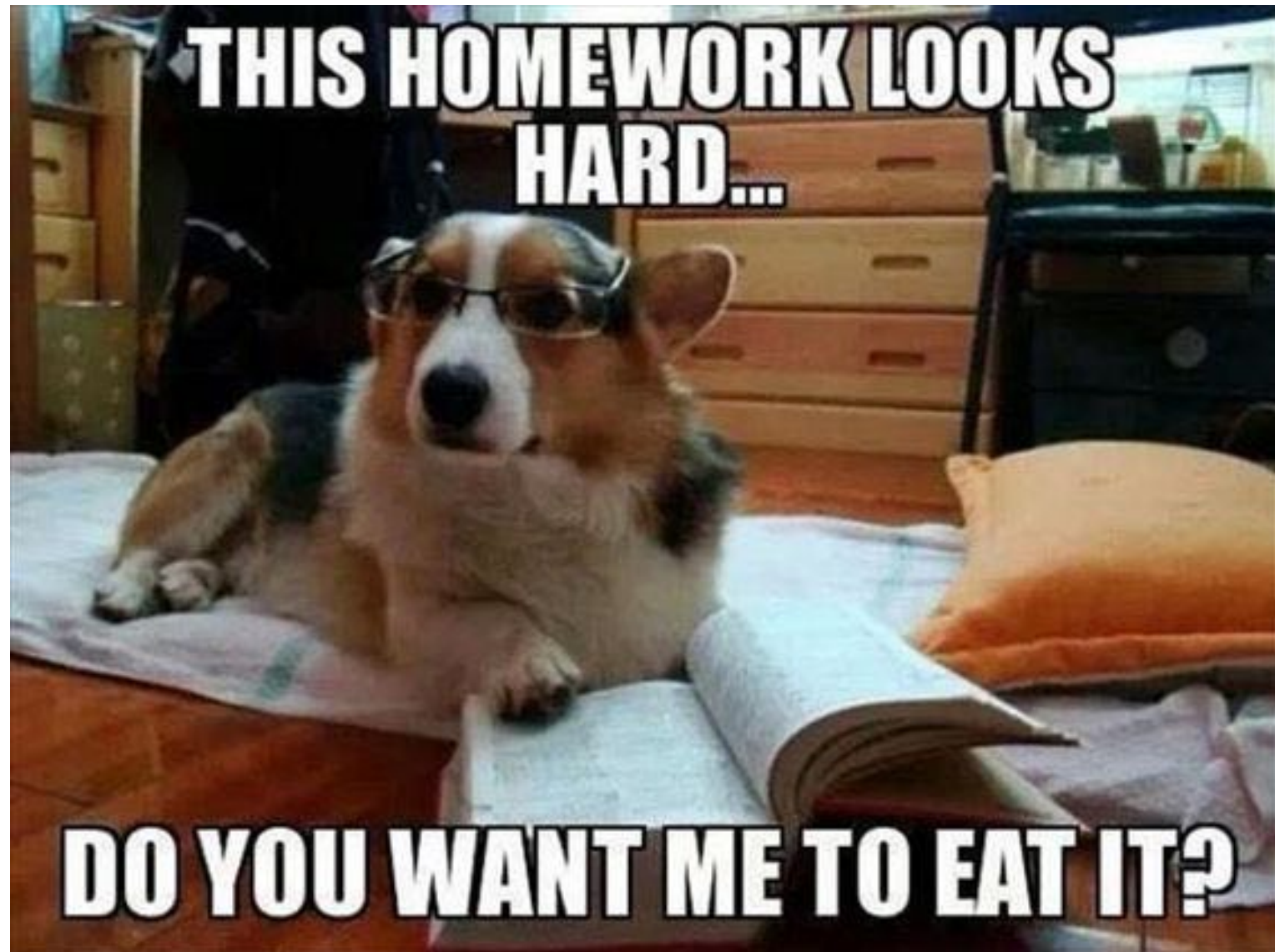
- Jon Duckett's books *HTML & CSS* and *JavaScript & jQuery*.
- These are two of the clearest primers on their respective subjects. If you're a "book-person," these may be helpful texts.

Feedback #2 – Books Please?



- ***Eloquent JavaScript***: a terrific, free book that goes through the fundamentals of JavaScript.
- You can find it here:
<http://eloquentjavascript.net/>

Feedback #3 – Homework is Hard



Feedback #3 – Homework is Hard

- **Check your homework due dates in Bootcamp Spot.**
- Generally speaking, expect homework to be semi-challenging.
- Make sure you're leaving a solid block of time to put in.
- *If you find yourself working on assignment for over 2 hours without even a small amount of progress, **let us know!***
- You don't win points in life staring at a blank screen.

Recapping

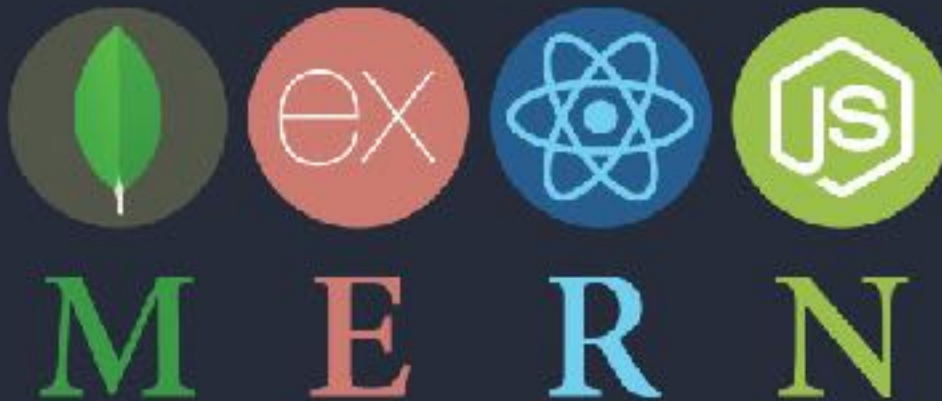
Recap

In just one whirlwind week we've covered:

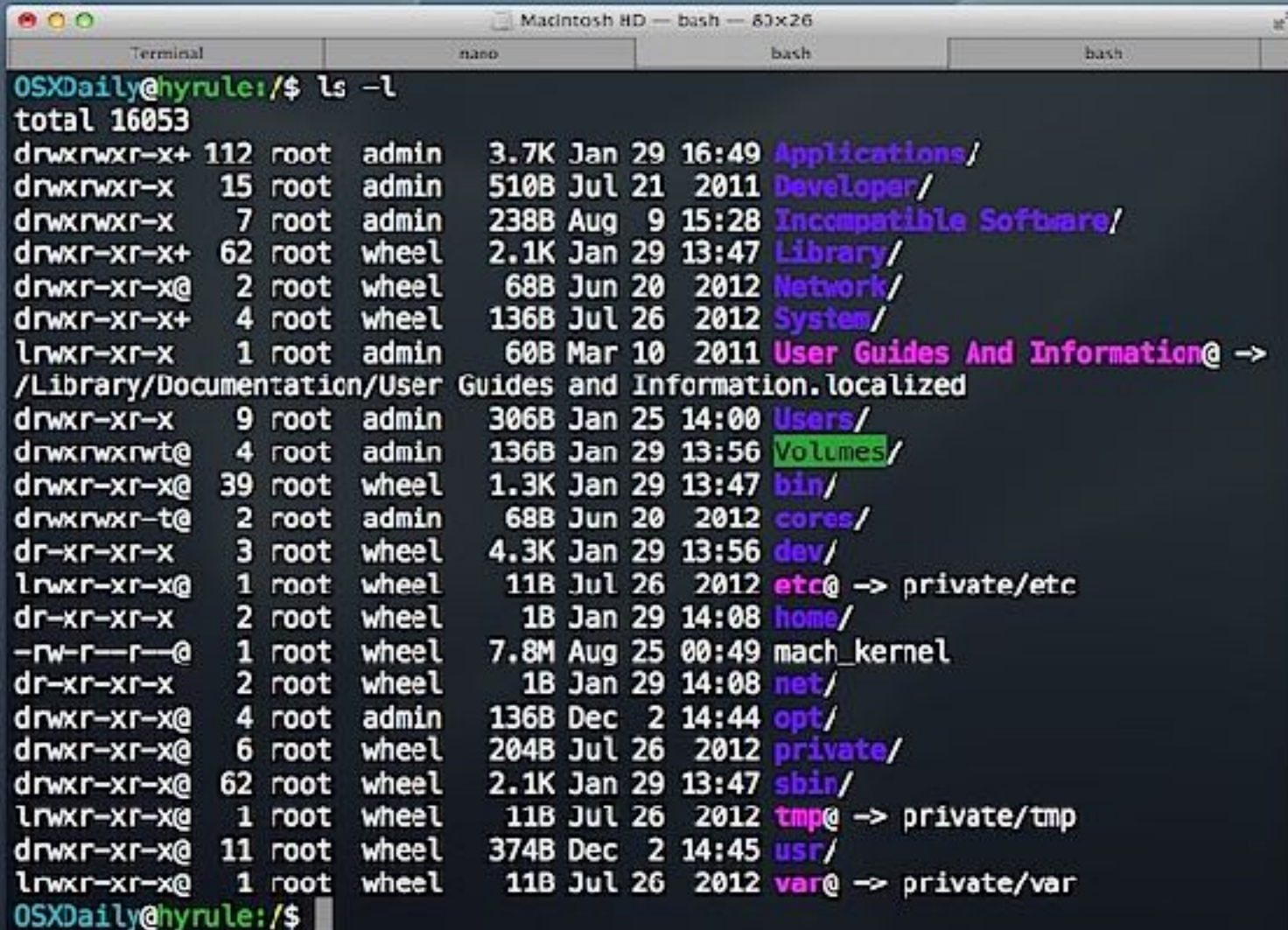
- Full-Stack Development Conceptually
- Terminal / Git Bash
- HTML Syntax
- Git Concepts and Commands
- CSS Purpose, Syntax, and Styles
- Floating
- Positioning
- Box Model
- Chrome Dev Tools
- **How to Learn on Your Own!!**



Full-Stack Development?



> Intro to Console



A screenshot of a macOS Terminal window titled "Macintosh HD — bash — 80x26". The window shows the output of the command `ls -l` executed by the user `OSXDaily@hyrule:/`. The output lists the contents of the root directory with detailed permissions, ownership, size, and timestamps. The files and directories listed are: `Applications/`, `Developer/`, `Incompatible Software/`, `Library/`, `Network/`, `System/`, `User Guides And Information@ -> /Library/Documentation/User Guides and Information.localized`, `Users/`, `Volumes/` (highlighted in green), `bin/`, `cores/`, `dev/`, `etc@ -> private/etc`, `home/`, `mach_kernel`, `net/`, `opt/`, `private/`, `sbin/`, `tmp@ -> private/tmp`, `usr/`, and `var@ -> private/var`. The prompt `OSXDaily@hyrule:/` is shown at the bottom.

```
OSXDaily@hyrule:/ $ ls -l
total 16053
drwxrwxr-x+ 112 root  admin   3.7K Jan 29 16:49 Applications/
drwxrwxr-x   15 root  admin   510B Jul 21  2011 Developer/
drwxrwxr-x    7 root  admin   238B Aug  9 15:28 Incompatible Software/
drwxr-xr-x+  62 root  wheel   2.1K Jan 29 13:47 Library/
drwxr-xr-x@   2 root  wheel    68B Jun 20  2012 Network/
drwxr-xr-x+   4 root  wheel   136B Jul 26  2012 System/
lrwxr-xr-x    1 root  admin    60B Mar 10  2011 User Guides And Information@ ->
/Library/Documentation/User Guides and Information.localized
drwxr-xr-x    9 root  admin   306B Jan 25 14:00 Users/
drwxrwxrwt@   4 root  admin   136B Jan 29 13:56 Volumes/
drwxr-xr-x@  39 root  wheel   1.3K Jan 29 13:47 bin/
drwxrwxr-t@   2 root  admin    68B Jun 20  2012 cores/
dr-xr-xr-x    3 root  wheel   4.3K Jan 29 13:56 dev/
lrwxr-xr-x@   1 root  wheel    11B Jul 26  2012 etc@ -> private/etc
dr-xr-xr-x    2 root  wheel     1B Jan 29 14:08 home/
-rw-r--r--@   1 root  wheel   7.8M Aug 25 00:49 mach_kernel
dr-xr-xr-x    2 root  wheel     1B Jan 29 14:08 net/
drwxr-xr-x@   4 root  admin   136B Dec  2 14:44 opt/
drwxr-xr-x@   6 root  wheel   204B Jul 26  2012 private/
drwxr-xr-x@  62 root  wheel   2.1K Jan 29 13:47 sbin/
lrwxr-xr-x@   1 root  wheel    11B Jul 26  2012 tmp@ -> private/tmp
drwxr-xr-x@  11 root  wheel   374B Dec  2 14:45 usr/
lrwxr-xr-x@   1 root  wheel    11B Jul 26  2012 var@ -> private/var
OSXDaily@hyrule:/ $
```


<title> Intro to HTML </title>



- **HTML** is one of the three base languages behind every single website.
- It defines all of the basic content and a *bit* of formatting.

Pushing and Pulling to GitHub



1

2

3

4

GitHub Branch



Pull Code

Push Code



Pull Code

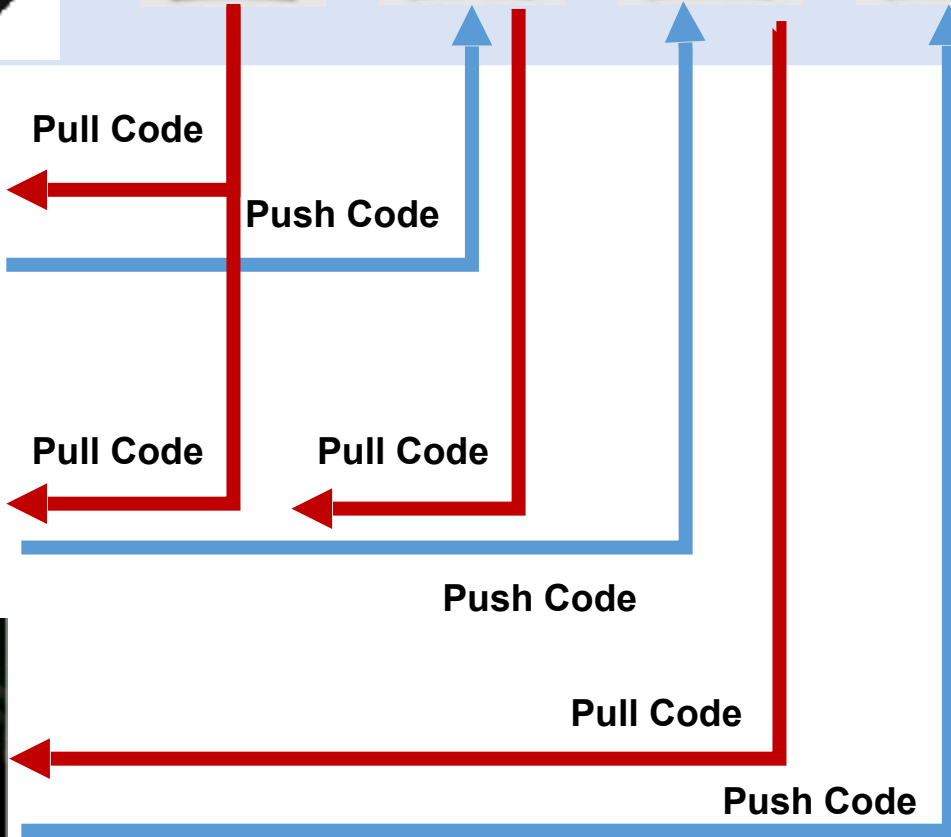
Pull Code

Push Code



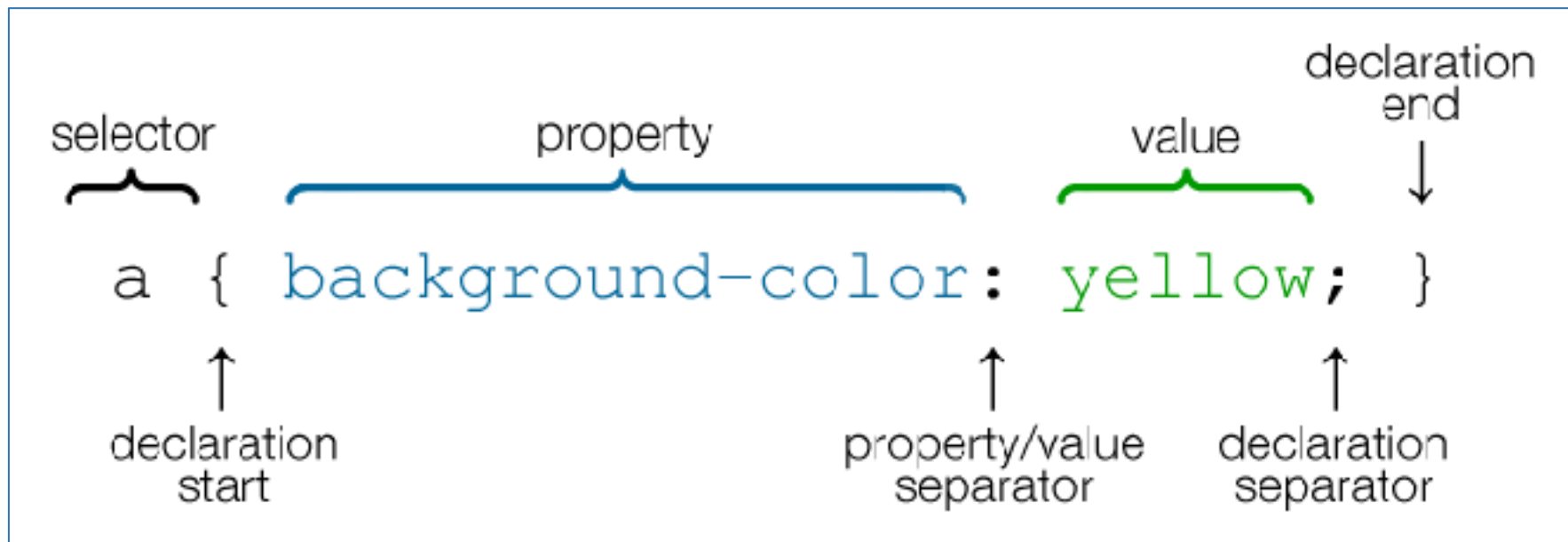
Pull Code

Push Code

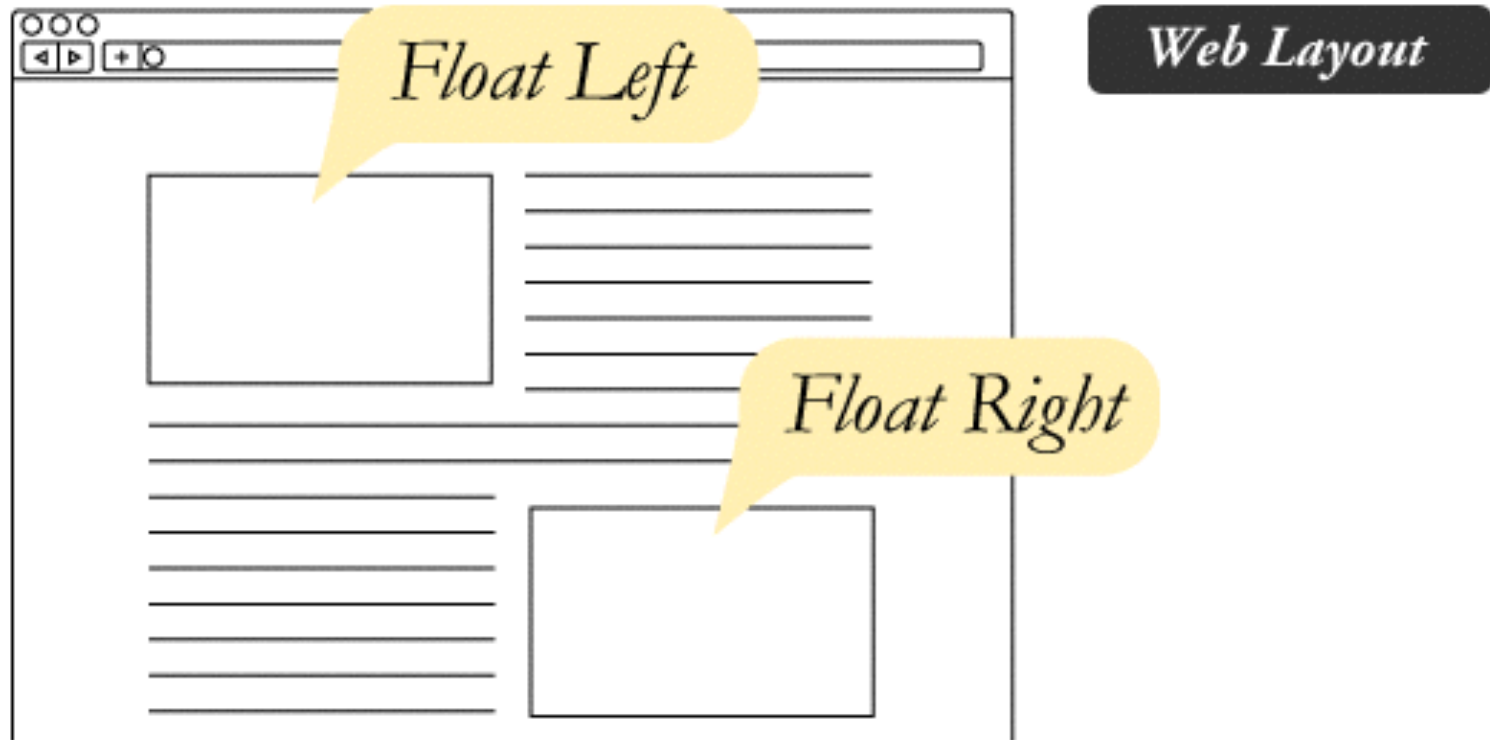


CSS Syntax

- CSS works by hooking onto **selectors** added into HTML using “**classes** and **identifiers**”.
- Once hooked, we apply **styles** to those HTML elements using CSS.

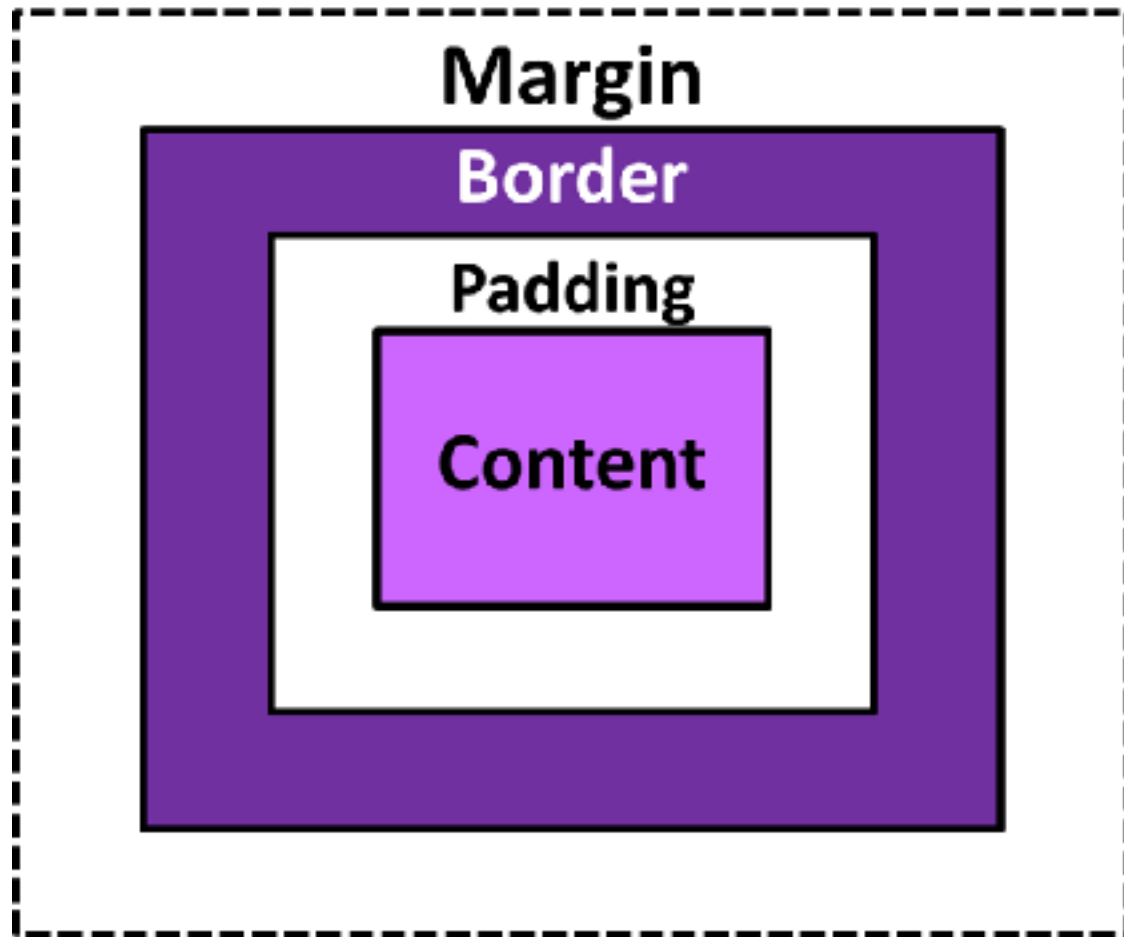


The Concept of “Flow”



- In HTML/CSS, (by default) every element displayed is governed by a concept called “**flow**.”
- This means that HTML elements force their adjacent elements to **flow around** them.

The Box Model



- The Box Model wraps every CSS element in **padding, border and margin** – allowing developers to modify spacing styles.

How to Learn...

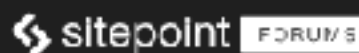


design shack



stackoverflow

CSS-TRICKS



SMASHING
MAGAZINE



MDN
MOZILLA
DEVELOPER
NETWORK

General Questions / Issues?



Double Take

Divs, Sections, Navs, Etc....



What is the difference between `<section>` and `<div>`?

Are new HTML5 elements like `<section>` and `<article>` pointless? [closed]

Why to use HTML5 semantic tag instead of div [duplicate]

10 months ago by Noel Hale

USE DIV OR SECTION ELEMENT

Section (Layout) vs. Div Block (Basic): Which one to use for Semantic Markup and Page Layout?

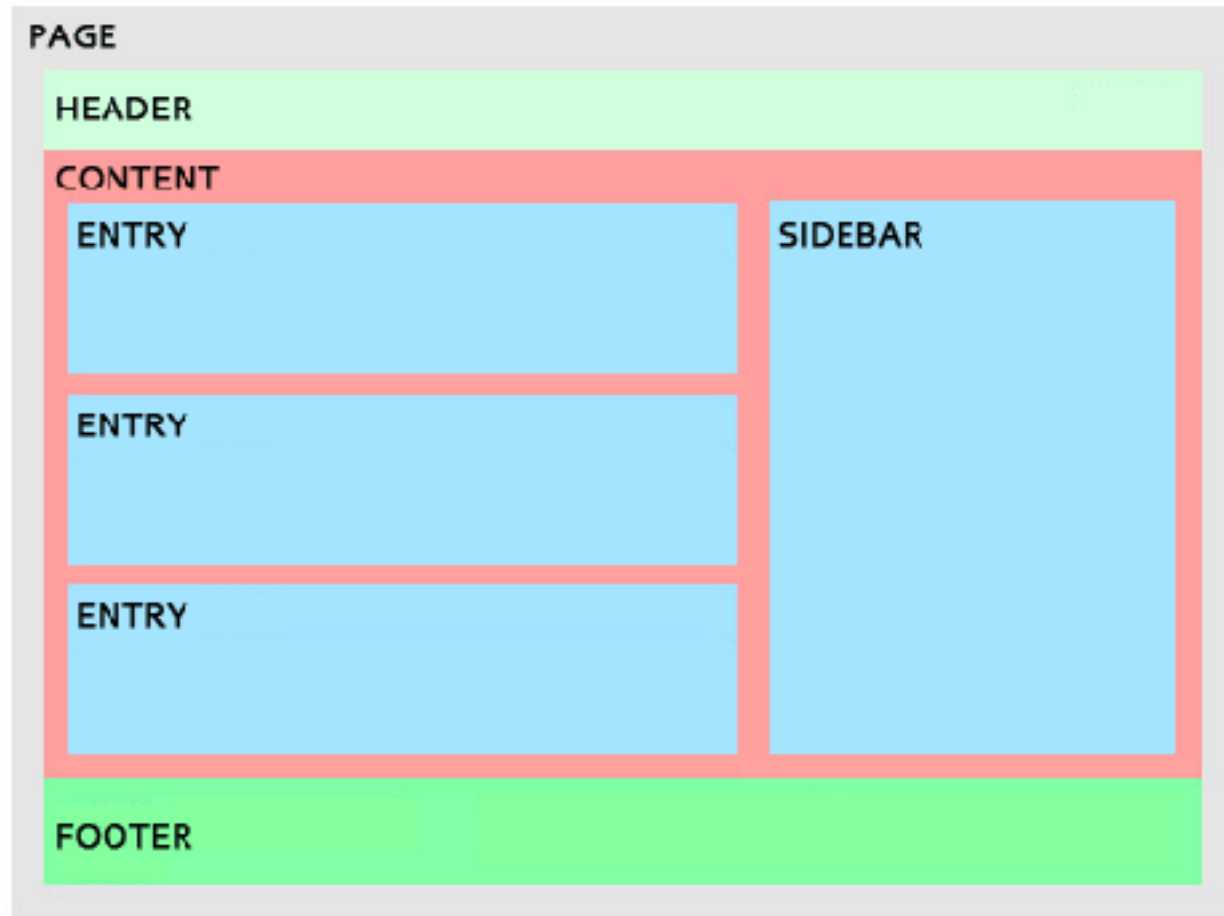
■ Need Help

About.com > About Tech > Web Design & HTML > ... > HTML5 Tags

What is the Difference Between DIV and SECTION?

Understanding the HTML5 SECTION Element

Divs, Sections, Navs, Etc....

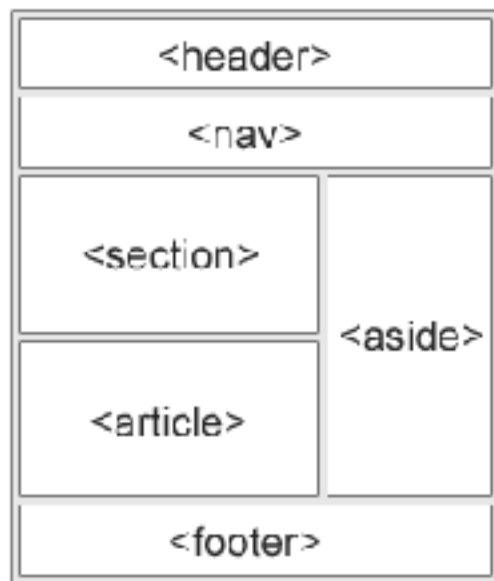


- All web layouts are inherently composed of containers, traditionally called “**divs**.”

Divs, Sections, Navs, Etc....

Website Layout Using HTML5

HTML5 offers new semantic elements that define different parts of a web page:



- `<header>` Defines a header for a document or a section
- `<nav>` - Defines a container for navigation links
- `<section>` Defines a section in a document
- `<article>` - Defines an independent self-contained article
- `<aside>` Defines content aside from the content (like a sidebar)
- `<footer>` - Defines a footer for a document or a section
- `<details>` Defines additional details
- `<summary>` - Defines a heading for the `<details>` element

- HTML5 introduced the concept of “**semantic layouts**,” meaning “divs” could be given more meaningful names.
- In theory, this helps with organization and search engine optimization.

Divs, Sections, Navs, Etc....

w3schools.com

[HTML](#) [CSS](#) [JAVASCRIPT](#) [SQL](#) [PHP](#) [BOOTSTRAP](#) [JQUERY](#) [ANGULAR](#) [XML](#)

HTML Tutorial

HTML HOME

HTML Introduction

HTML Editors

HTML Basic

HTML Elements

HTML Attributes

HTML Headings

HTML Paragraphs

HTML Styles

HTML Formatting

HTML Quotations

HTML Computercode

HTML Comments

node

Platform for developer focus and flow

Fast deploys, choice of version, MongoDB

SIGN UP FOR FREE

HTML5 Semantic Elements

« Previous

Semantics is the study of the meanings of words and phrases in language.

Semantic elements are elements with a meaning.

What are Semantic Elements?

Elements Console Sources Network Timeline Profiles Resources Security Audits

```
<!DOCTYPE html>
<html lang="en-US">
  <head>...</head>
  <body>
    <div class="w3-container w3-white">...</div>
    <div class="w3-topnav w3-card-2 w3-slim topnav" id="topnav" style="position: relative; top: 0px;">...</div>
    <div class="w3-row w3-light-grey" id="outertopnav" style="padding-top: 0px;">
      <div class="w3-col w3-slim" id="leftmenu" style="padding-top: 0px; display: none;">...</div>
      <div class="w3-rest">
        <div class="w3-row w3-white">
          <div class="w3-col w3-card-2" id="main">...</div>
          <div class="w3-col w3-card-2" id="right">...</div>
        </div>
        <div class="footer w3-container w3-white">...</div>
      </div>
    </div>
    <div id="nav_tutorials_content" style="display:none;">...</div>
    <div id="nav_references_content" style="display:none;">...</div>
    <div id="nav_examples_content" style="display:none;">...</div>
    <div id="nav_translat_content" style="display:none;">...</div>
    <div id="nav_search_content" style="display:none;">...</div>
    <script src="http://www.w3schools.com/js/4/jquery.js"></script>
  </body>
</html>
<script src="http://www.w3schools.com/js/4/jquery.js"></script>
```

- That said... many (if not most) websites, seem to still be using basic **divs**.

- There are reasons for this that we'll showcase in later sections.

- Additionally, it's possible to include “semantics” by using id names and classes.

Divs, Sections, Navs, Etc....

div?

Section?

- **Bottom line:**

Follow your homework's instructions. But when you get out in the “real world,” follow the convention of where you work!

Classes vs. IDs

Classes = Barcode (all iPod)



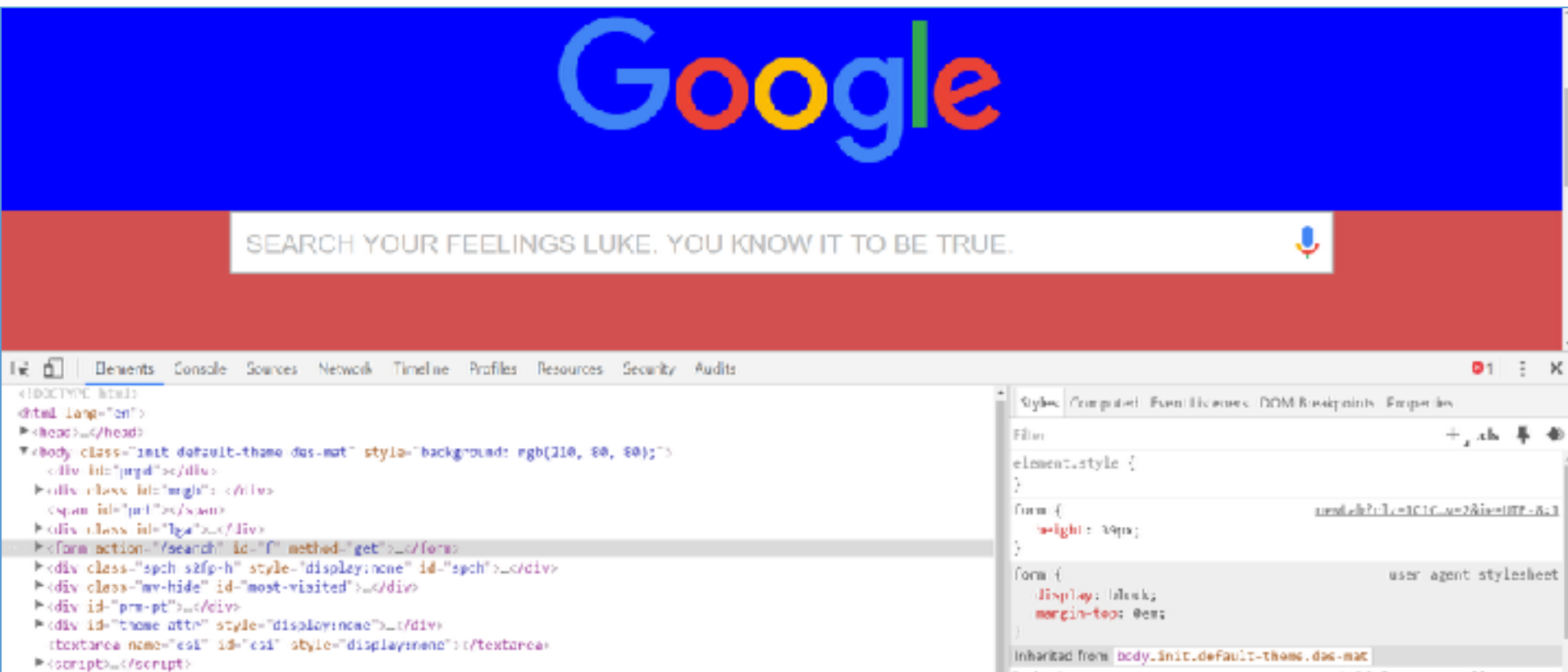
IDs = Serial Number (unique iPod)



When choosing between a CSS ID and a CSS Class follow the convention:

- **Classes (.classname)** are to be used if the same style will be used on multiple HTML elements.
- **IDs (#idname)** are to be used if a style is *unique* to that HTML element.

Modifying Sites



- You can edit any page's HTML and CSS with Chrome Developer Tools.
- Plus, you'll see your results instantly.

Practice Through Frustration

Keep Practicing!
It gets better.

CSS Typography

Typography is Huge



- **Typography** can play a huge role in the visual aesthetic and emotional identity of a webpage.

Typographic Properties

- **Line Height**

Distance between lines of text on a page (Golden Ratio of 1.5x font).

- **Font-Size**

Actual size of lettering. Should be at least 16 px on modern pages.

<https://www.smashingmagazine.com/2011/10/16-pixels-body-copy-anything-less-costly-mistake>

- **Line Length**

Not a CSS property but rather a standard. Should be about 50-75 characters per line on a desktop .

<http://baymard.com/blog/line-length-readability>

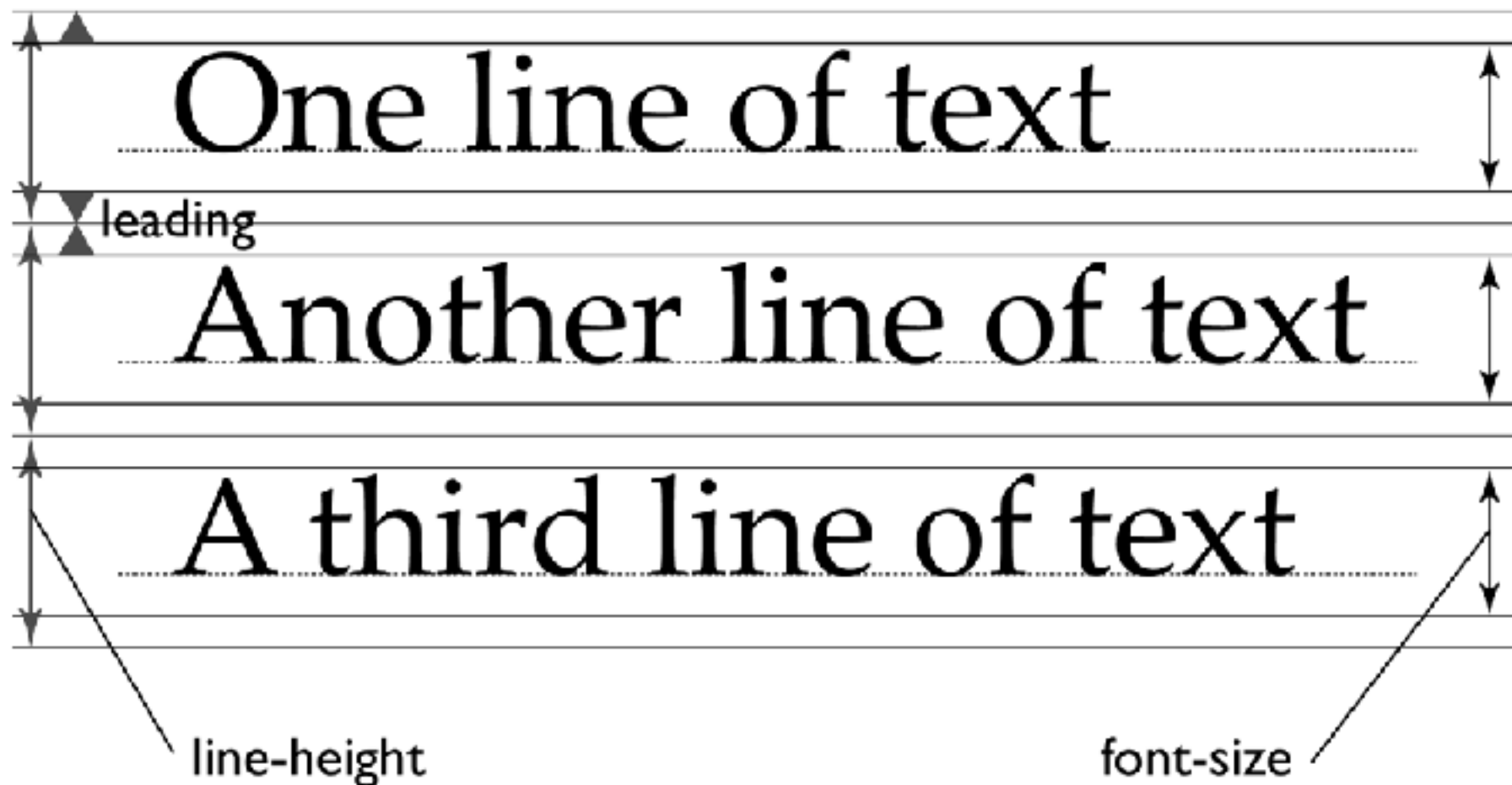
- **Letter Spacing**

Spacing between individual letters—try to avoid cramping.

- **Sans-Serif vs Serif**

Sans-serif are fonts sans their serifs—no tails. They're easier to read online.

Line Height vs Fonts



Serif vs Sans Serif



- **Serif fonts** include small lines attached to the end strokes of letters.
- **Sans-Serif** (without serif) do not include such strokes.

> YOUR TURN!!

Assignment

Create a block of text using a Lorem Ipsum Generator. Incorporate it into an HTML Document.

Then use CSS styling to modify the:

- Font-family
- Font-size
- Font-weight
- Line-height
- Letter-spacing

Bonus:

If you finish early, read about TypePlate and see if you can figure out how to incorporate it into your site (<http://typeplate.com/>).

Google Fonts

The screenshot displays the Google Fonts website interface. At the top, the Google Fonts logo is on the left, and links for 'More scripts', 'About', 'Analytics', and 'New to Google Fonts?' are on the right. Below the header, a search bar shows '708 font families shown'. To the right of the search bar are tabs for 'Word', 'Sentence', 'Paragraph', and 'Poster', with 'Sentence' selected. Further right are icons for 'CC' and a download arrow. Below the search bar, a 'Preview Text' field contains 'Grumpy wizards make toxic brew for the evil', followed by a 'Size' dropdown set to '28 px'. On the left side, under 'Filters:', there is a dropdown for 'All categories' and a list of filter categories: 'Thickness', 'Slant', and 'Width'. Below the filters, a 'Script:' dropdown is set to 'Latin', and a checkbox for 'Reset all filters/search' is present. Under 'Styles:', there is a checkbox for 'Show all styles'. The main preview area shows two font examples. The first example is 'Open Sans, 10 Styles by Steve Matteson', with a preview of the text 'Grumpy wizards make toxic brew for the evil Q' in a sans-serif font. The second example is 'Roboto, 12 Styles by Christian Robertson', with a preview of the text 'Grumpy wizards make toxic brew for the evil Que' in a sans-serif font. Both examples include icons for 'Add to Collection' and a 'Download' button.

Google Fonts

More scripts About Analytics New to Google Fonts?

708 font families shown

Word Sentence Paragraph Poster

Preview Text: Grumpy wizards make toxic brew for the evil

Size: 28 px

Filters:

All categories

Thickness

Slant

Width

Script:

Latin

☐ Reset all filters/search

Styles:

☐ Show all styles

Normal 100

Grumpy wizards make toxic brew for the evil Q

Open Sans, 10 Styles by Steve Matteson

Add to Collection

Normal 100

Grumpy wizards make toxic brew for the evil Que

Roboto, 12 Styles by Christian Robertson

Add to Collection

You can easily incorporate Google's custom fonts.

Google Fonts

> YOUR TURN!!

Assignment:

Create (or re-use) an HTML Document of your choosing and then incorporate 2-3 custom Google fonts into the page.

Bonus:

If you finish early, read about Font Awesome icons and see if you can figure out how to incorporate one onto your page.

Lunch (30 mins)

Pseudo Styles

Pseudo Classes

- CSS has keywords that can be added to selectors. These highlight the **special states** of the selected element.
- Great example: CSS can hook onto the specific moment when a link is **hovered over** using the **:hover** pseudo-class.
- Complete list of pseudo-classes found here: <https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes>

Index of standard pseudo-classes

<code>:active</code>	<code>:link</code>
<code>:any</code>	<code>:not()</code>
<code>:checked</code>	<code>:nth-child()</code>
<code>:default</code>	<code>:nth-last-child()</code>
<code>:dir()</code>	<code>:nth-last-of-type()</code>
<code>:disabled</code>	<code>:nth-of-type()</code>
<code>:empty</code>	<code>:only-child</code>
<code>:enabled</code>	<code>:only-of-type</code>
<code>:first</code>	<code>:optional</code>
<code>:first-child</code>	<code>:out-of-range</code>
<code>:first-of-type</code>	<code>:read-only</code>
<code>:fullscreen</code>	<code>:read-write</code>
<code>:focus</code>	<code>:required</code>
<code>:hover</code>	<code>:right</code>
<code>:indeterminate</code>	<code>:root</code>
<code>:in-range</code>	<code>:scope</code>
<code>:invalid</code>	<code>:target</code>
<code>:lang()</code>	<code>:valid</code>
<code>:last-child</code>	<code>:visited</code>
<code>:last-of-type</code>	
<code>:left</code>	

Button with Various States

Default State

Hover State

Focus State

Active State

Instructor: Demo
(Button.html | 9-Pseudoclass)

> YOUR TURN!!

Assignment

Create a series of links and modify the pseudo-classes associated with their Default, Active, Hover and Focus States.

Bonus:

If you finish early, incorporate a pseudo-class we haven't covered in your page. Then read about “pseudo-elements” and try to incorporate one as well.

Beyond CSS Hacking

Quick Refresher on HTML/CSS

- **HTML:** Hypertext Markup Language – (Content)
- **CSS:** Cascading Style Sheets – (Appearance)
- **HTML/CSS are the “languages of the web”.** Together they define both the content and the aesthetics of a webpage – handling everything from the layouts, colors, fonts, and content placement. (Javascript is the third – handling logic, animation, etc.)



HTML vs HTML/CSS

HTML Alone

- Like writing papers in “Notepad.”
- Can only write unformatted text.



HTML / CSS

- Like writing papers in Microsoft Word.
- Can format text, page settings, alignment, etc. based on “highlighting” and menu options.



How You (Probably) Feel About CSS

CSS in a nutshell



A Revelation...

There is a better way!

Where's the CSS? (Instructor Demo)

[The Coding Bootcamp](#)[Home](#)[About](#)[Contact](#)[Dropdown ▾](#)[Default](#)[Static top](#)[Fixed top](#)

Awesome Header

Smaller Awesome Header

Even Smaller Header

[✉ Learn more](#)

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quis, molestiae modi accusantium animi magnam veniam, imperdiet asperiores sint consequuntur dictas fugit quibusdam dolor, iusto sed porro ad minus voluptatibus amet



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Accusamus, maiores, quos. At, animi, tacites. Aliis itaque ab minima nam, utaque animi nesciunt. Nam nesciunt est cumque voluptas, qui reprehenderit atque

> YOUR TURN!!

Quick Activity:

With those closest to you, take a close look at the Bootstrap-powered HTML document I just sent you.

Be prepared to answer the following two questions:

1. Where does this HTML document draw its styles from?
2. How does our browser know which HTML element to style, and in which way? (ex: button face or background banner)

HINT: Think about how Google Fonts work.

Powered by Bootstrap

External CSS Link

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>New York Times Search</title>
  <!-- Bootstrap CSS CDN -->
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.0.0/css/bootstrap.min.css" />
  <!-- Font Awesome CSS Icons (For cool glyphs) -->
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.5.0/css/font-awesome.min.css" />
</body>
```

Pre-Defined CSS Class (ex: “Jumbotron”)

```
<!-- Jumbotron for Title -->
<div class="jumbotron">
  <h1 class="text-center">
    <strong>
      <i class="fa fa-newspaper-o"></i> New York Times Search</strong>
    </h1>
</div>
```


Powered by Bootstrap

The Coding Bootcamp

Home

About

Contact

Dropdown ▾

Default

Static top

Fixed top

Awesome Header

Smaller Awesome Header

Even Smaller Header

✉ Learn more

class="jumbotron"

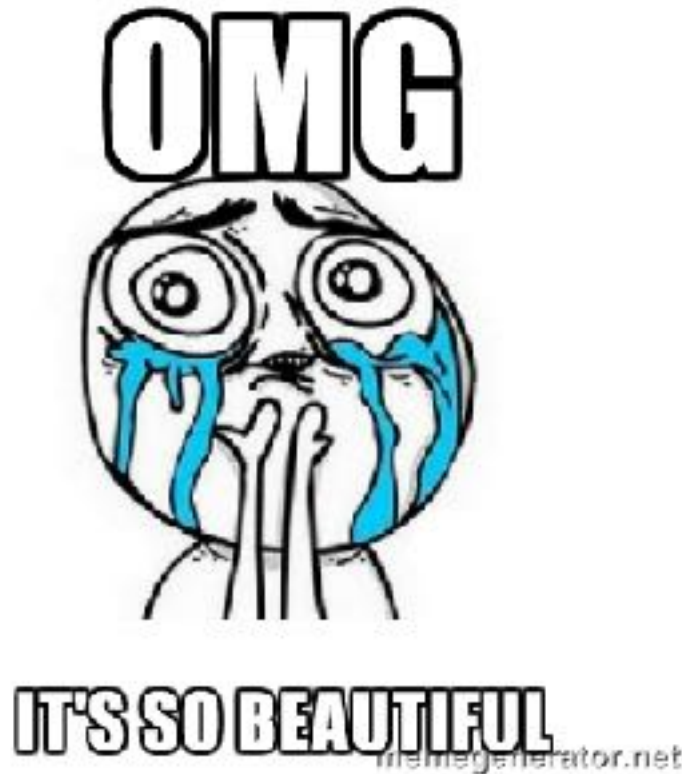


Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quis, molestiae modi accusantium animi magnam veniam, impedit asperiores sint consequatur debiles fugi quibusdam dolor, iusto sed porro ab minus voluptatibus amet



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Accusamus, maiores, quos. At, animi, tacites. Aliis, itaque sibi minima, necum, exequis animi insequunt. Itaque necum est cumque voluptas, qui reprehenderit atque

Big Deal!!!



***Using Bootstrap,
we have PRE-DEFINED CSS Styles.***

> YOUR TURN!!

Quick Activity:

Turn to the person next to you and celebrate this together.

Then discuss the concept of Bootstrap to confirm that you both understand why you are celebrating.

Hint: You should be explaining why is Bootstrap's "pre-defined" CSS a big deal.

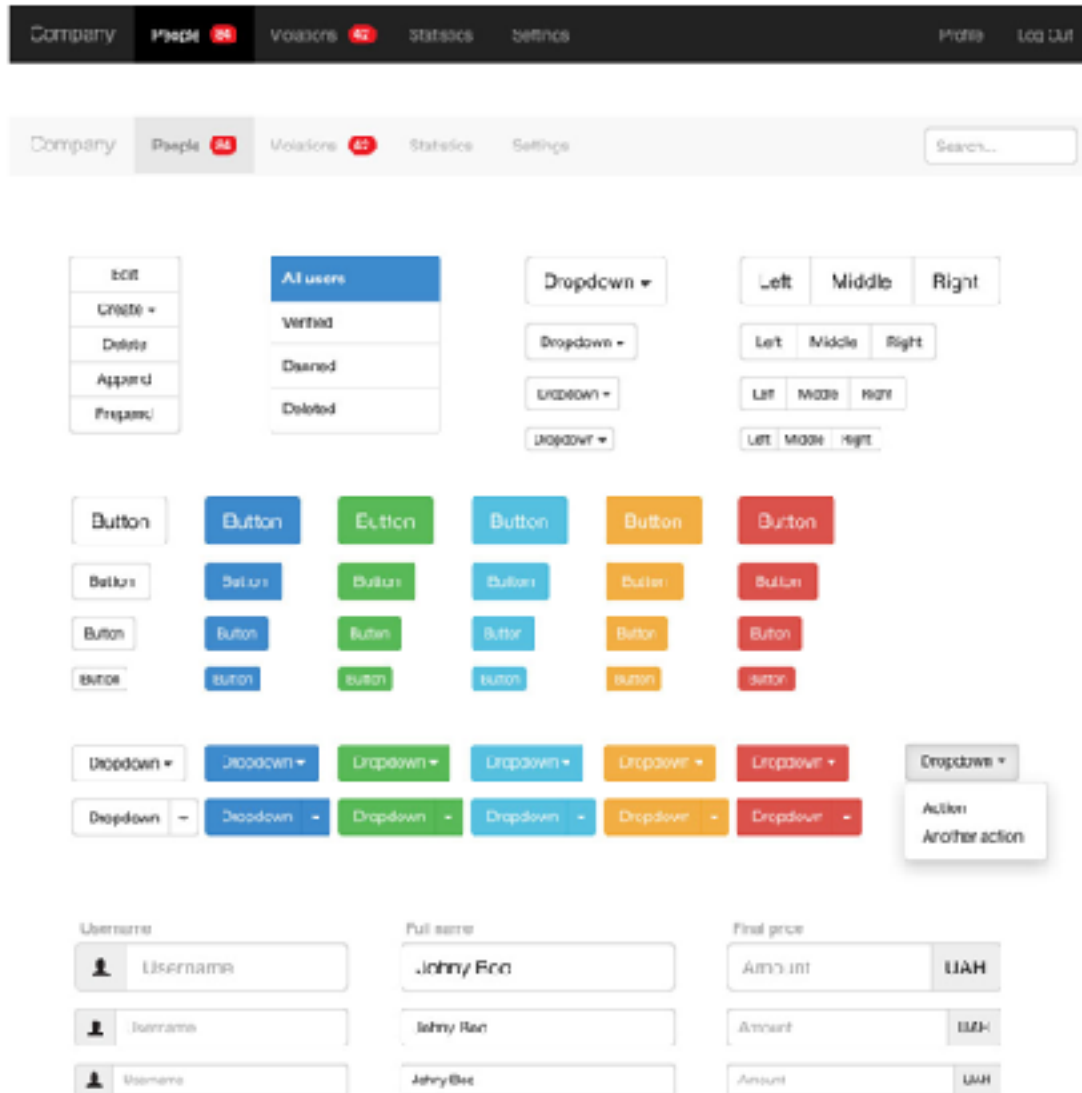
What is Bootstrap?

Bootstrap Definition

- **Twitter Bootstrap** is a free collection of tools for creating websites and web applications.
- It comes with a pre-built design template for typography, forms, buttons, navigation, UI elements and JavaScript.
- Documentation here: <http://getbootstrap.com/>



Why Use Bootstrap?



- **Reason #1: UI Kit**
- Familiarize yourself with the UI features it offers via the documentation.
- Once Bootstrap is active, you can simply copy snippets from the documentation to save yourself major time of creating elements yourself.

Why Use Bootstrap?

Reason #2: Mobile Responsiveness

- One of the most compelling reasons to use Bootstrap is the default **mobile-responsive** quality it provides.
- This means that your website will look “good” automatically when viewed on screens ranging from monitors to tablets to phones.



HTML / CSS / Bootstrap Analogy

HTML Alone

- Like writing papers in “Notepad.”
- Can only write unformatted text.



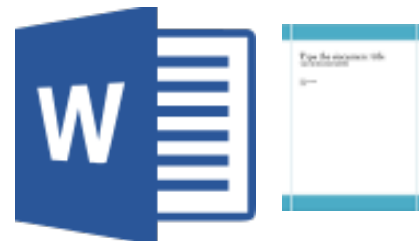
HTML / CSS

- Like writing papers in Microsoft Word.
- Can format text, page settings, alignment, etc. based on “highlighting” and menu options.



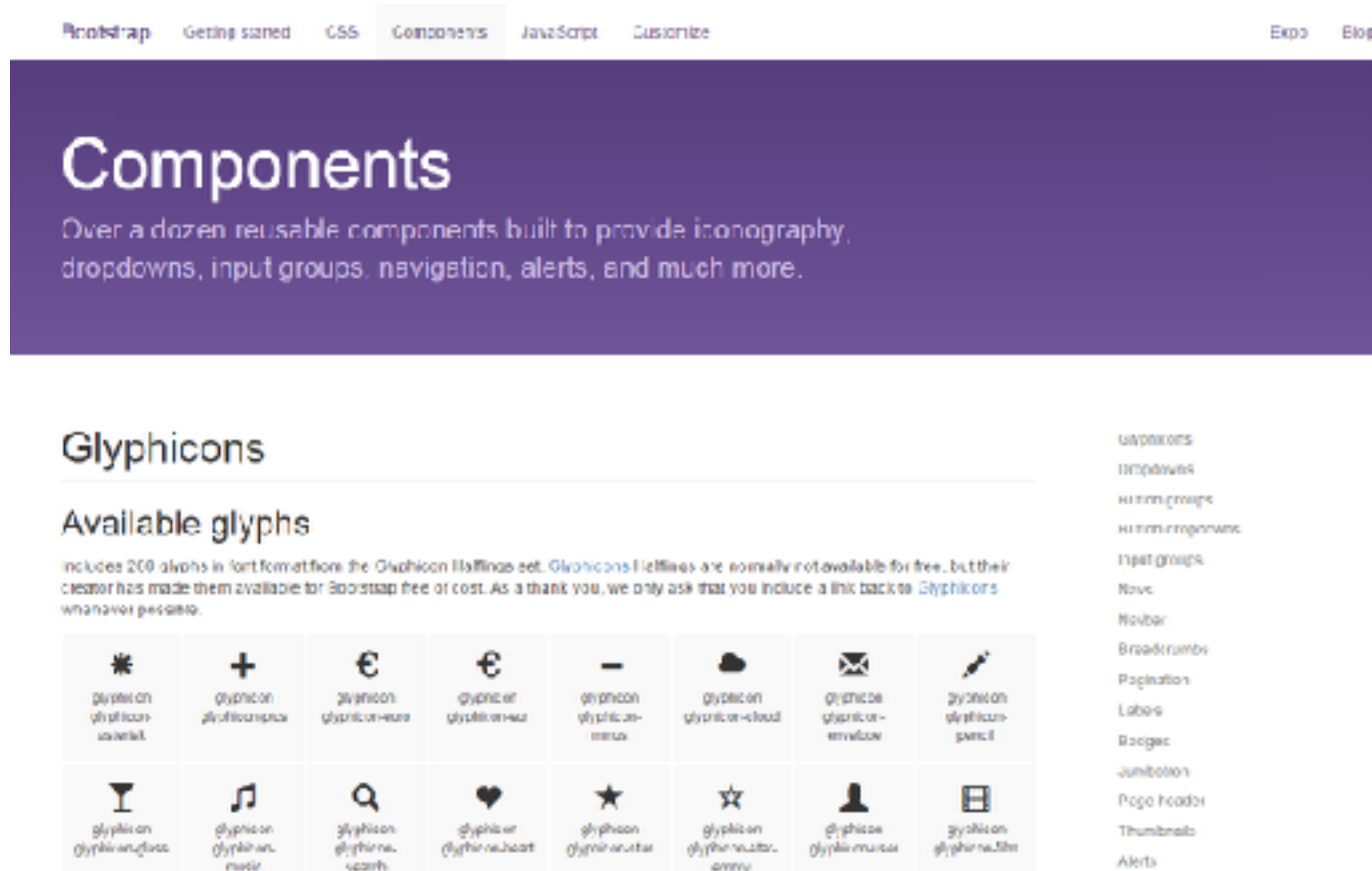
HTML / CSS & Bootstrap

- Like writing papers in Microsoft Word with a **prebuilt template**.
- You can still customize, but now have a pre-built style and aesthetic look.



Key Bootstrap Features

Bootstrap Components



- **Bootstrap** offers a wide range of components that you should consider including in your next web projects. Flip through the documentation and incorporate elements you see fit.

Bootstrap Nav Bars

My First Website Link 1 Link 2 Link 3

Example:

```
<nav class="navbar navbar-default navbar-fixed-top" role="navigation">
  <div class="container">
    <div class="navbar header">
      <button type="button" class="navbar-toggle collapsed"
        data-toggle="collapse" data-target="#navbar" aria-expanded="false" aria-controls="navbar">
        <span class="sr-only">toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="page-scroll navbar-brand" href="#page-top">My First Website</a>
    </div> <!-- /.navbar header -->
    <div id="navbar" class="navbar-collapse collapse">
      <ul class="nav navbar-nav">
        <li class="hidden">
          <a href="#page-top"></a>
        </li>
        <li><a href="http://www.google.com" class="nav-btn">Link 1</a></li>
        <li><a href="http://www.facebook.com" class="nav-btn">Link 2</a></li>
        <li><a href="http://www.twitter.com" class="nav-btn">Link 3</a></li>
      </ul>
    </div> <!-- /.navbar collapse -->
  </div> <!-- /.container -->
</nav>
```

Bootstrap Jumbotron

Hello, world!

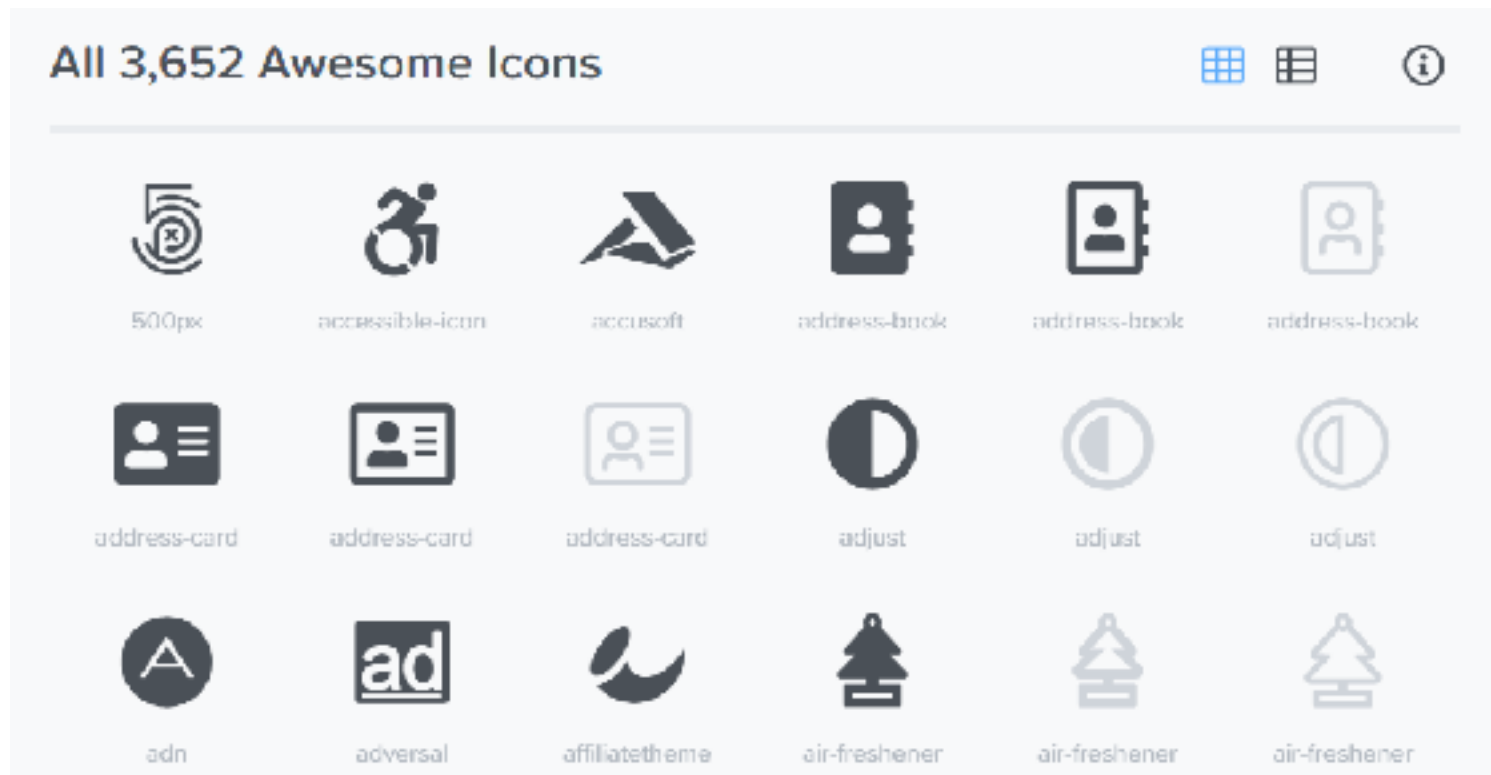
This is a simple hero unit, a simple jumbotron-style component for calling extra attention to featured content or information.

Learn more

Example:

```
<div class="jumbotron">
  <h1>Hello, world!</h1>
  <p>...</p>
  <p><a class="btn btn-primary btn-lg" href="#" role="button">Learn more</a></p>
</div>
```

Bootstrap and Font Awesome



Example: ` Learn more`

Bootstrap Table

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

Example:

```
<table class="table table striped">
  <tr>
    <th>First Name</th>
    <th>Last Name</th>
    <th>Username</th>
  </tr>
  <tr>
    <td>Mark</td>
    <td>Otto</td>
    </td>@mdo</td>
  </tr>
  ...
</table>
```

Bootstrap Form

Email address

Password

File input

 No file chosen

Example block level help text here.

☐ Check me out

Example:

```
<form>
  <div class="form-group">
    <label for="exampleInputEmail">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail" placeholder="Enter email">
  </div>
  <div class="form-group">
    <label for="exampleInputPassword">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword" placeholder="Password">
  </div>
  <div class="form-group">
    <label for="exampleInputFile">File input</label>
    <input type="file" id="exampleInputFile">
    <p class="help-block">Example block level help text here.</p>
  </div>
  <div class="checkbox">
    <label>
      <input type="checkbox"> Check me out
    </label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```

Instructor: Demo
(BootstrapButtons.html | 11-BootstrapDemo)

> YOUR TURN!!

Activity:

Working with someone closest to you:

1. Navigate to the Twitter Bootstrap website.
2. Then copy the link to the Bootstrap CSS file into one of your old HTML files.
3. Look through the Bootstrap CSS or Components list and incorporate at least three Bootstrap elements onto your page.
4. Slack a screenshot of your page to your section's channel when you finish.

Hint: First include a link to Bootstrap's CSS. Then visit the Bootstrap page on CSS or Components.

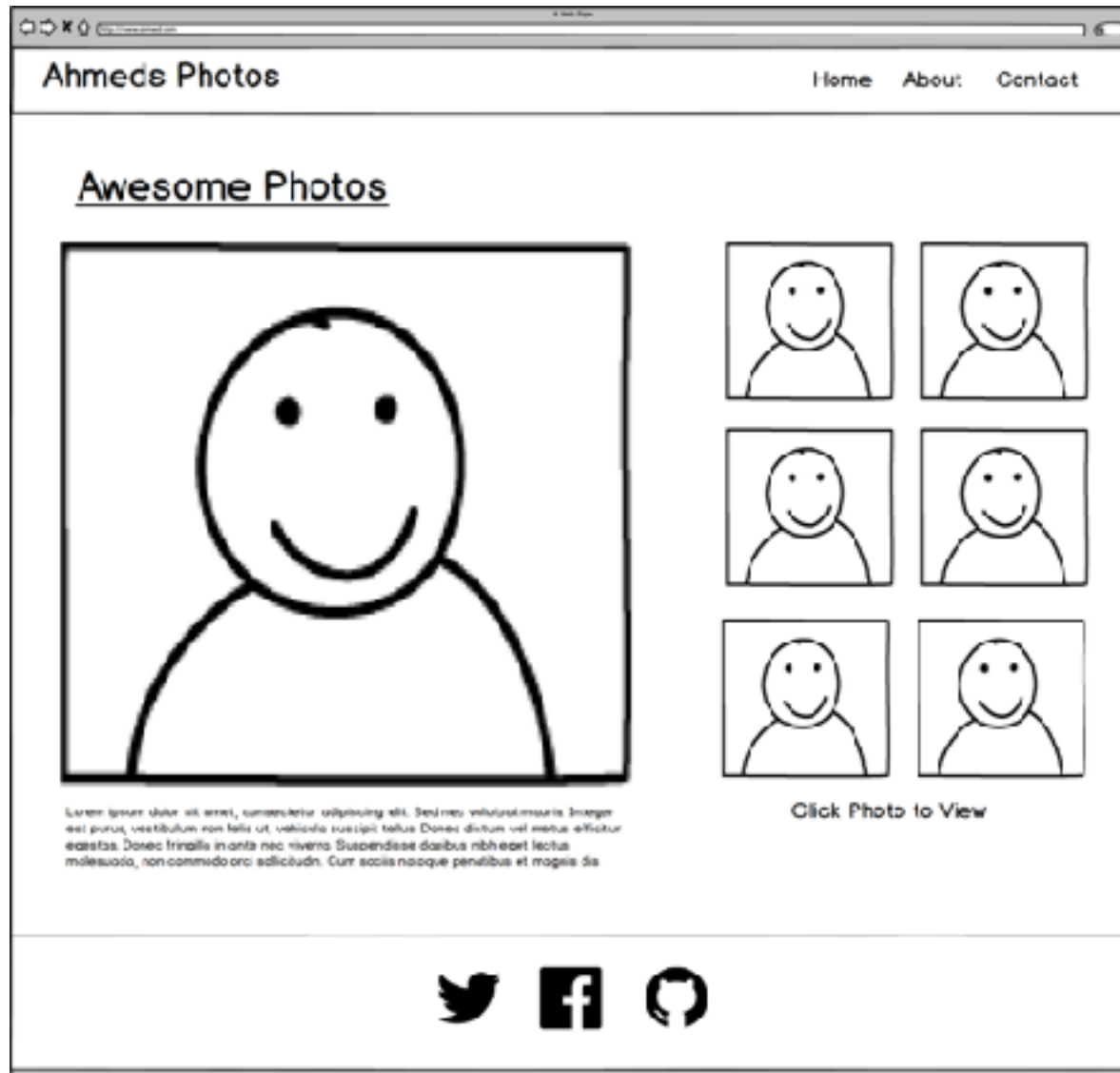
Takeaways

Takeaways

By the end of class you should be able to:

1. Offer a loose definition of “typography” and a few CSS properties for styling it.
2. Offer a practical example of using pseudo-classes.
3. Understand *generally* what Twitter Bootstrap is and why pre-defined CSS frameworks are powerful.
4. Know how to incorporate the Twitter Bootstrap style library into your website.

Next Class... Layout Building!



Homework #2

Questions?
