Heroes of CSS

The Coding Bootcamp

Admin Work

Homework Assignment

- **Also, at this point everyone should have access to the class repository in GitHub.**

  <<<<CLASS REPOSITORY LINK HERE>>>>

- **Homework Assignment #1 is** Due:

  MW Class – Wednesday (<<<<UPDATE DATE>>>>)

  TTH Class – Thursday (<<<<UPDATE DATE>>>>)

© 2016 | Coding Boot Camp - All Rights Reserved

Homework Assignment

- **Really, work hard on this assignment!** This assignment introduces you to fundamental concepts that we'll be building the entire course-long.

- **Review In Class Material, *especially* Exercises:** <<<<<UPDATE LINK HERE>>>>>

- **Work with your peers!** It's much better than screaming at your computer alone.

- **Ask Questions on Slack!** Your peers, TAs, and Instructors are all here to help when they can.

© 2016 | Coding Boot Camp - All Rights Reserved

Most Important of All….

Just Submit SOMETHING (even if it seems pretty crummy)!

© 2016 | Coding Boot Camp - All Rights Reserved

Warning!

Brace Yourselves

Today is going to be a bit tough.

But trust us!

It will all look easy a few weeks from now.

© 2016 | Coding Boot Camp - All Rights Reserved

Brace Yourselves

Don’t expect to understand EVERYTHING at once.

Today is all about getting immersed.

© 2016 | Coding Boot Camp - All Rights Reserved

CSS Recap

Critical Question

What is "CSS?"

© 2016 | Coding Boot Camp - All Rights Reserved

HTML / CSS Definitions (*yawn* unimportant)

- **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.)

© 2016 | Coding Boot Camp - All Rights Reserved

HTML / CSS 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.

© 2016 | Coding Boot Camp - All Rights Reserved

Basic HTML Page - Result

Awesome Header

Smaller Awesome Header

Even Smaller Header

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quidem consequatur unde aut dolores odio hic, accusamus recusandae ipsam illum enim voluptatibus obcaecati totam tempora eum quod sapiente. Corporis, quidem, culpa?

Menu Links

- Google
- Facebook
- Twitter

© 2016 | Coding Boot Camp - All Rights Reserved

Enter CSS - Result

Awesome Header

Smaller Awesome Header

Even Smaller Header

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quidem consequatur unde aut dolores odio hic, accusamus recusandae ipsam illum enim voluptatibus obcaecati totam tempora eum quod sapiente. Corporis, quidem, culpa?

Menu Links

- Google
- Facebook
- Twitter

© 2016 | Coding Boot Camp - All Rights Reserved

Critical Question

How do we style HTML…

Elements?

Classes?

IDs?

© 2016 | Coding Boot Camp - All Rights Reserved

CSS Syntax

- CSS works by hooking onto **selectors** added into HTML using **classes** and **identifiers.**

- Classes use **.classname,** IDs use **#idname**, and elements use just their name.

- Once hooked, we apply **styles** to those HTML elements using CSS.

© 2016 | Coding Boot Camp - All Rights Reserved

Selectors

Element selector

Element name
(p, a, div, span, etc)

Applies to all <p> elements

```
p {
    background-color: blue;
}
```

Class Selector

Period (.) + variable name
(.myDiv, .phoneNumber, etc.)

Applies to all elements with class="classItem"

```
.classItem {
    background-color: orange;
}
```

ID Selector

Hash (#) + variable name
(#myDiv, #phoneNumber)

Applies to all elements with id="idItem"

```
#idItem {
    background-color: green;
}
```

© 2016 | Coding Boot Camp - All Rights Reserved

CSS Selectors

```
p {
   background-color: blue;
}

.classItem {
   background-color: orange;
}

#idItem {
   background-color: green;
}
```

+

```
<p>
   A paragraph with a blue background.
</p>
<div class="classItem">
   A div with an orange background.
</div>
<div id="idItem">
   A div with a green background.
</div>
```

A paragraph with a blue background.

A div with an orange background.

A div with a green background.

© 2016 | Coding Boot Camp - All Rights Reserved

Lingering Questions

Questions so far?

© 2016 | Coding Boot Camp - All Rights Reserved

Relative File Paths

Relative File Paths

```html
<!DOCTYPE html>
<html>
<head>
    <title>1.2.9 Exercise</title>

    <!-- This critical line points your HTML to the CSS file. Notice the "relative" pathway  -->
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
```

- **Relative file paths** connect us with other files in our working directory. In this case, style.css is in the same folder as our html document.

© 2016 | Coding Boot Camp - All Rights Reserved

Quick Demo

Instructor: Demo

(RelativePaths_DEMO | 1-RelativePaths)

© 2016 | Coding Boot Camp - All Rights Reserved

Absolutely No Absolute Paths

VERY <u>VERY</u> BAD

```
<!-- BAD!!!! -->
<link rel="stylesheet" href="D:/trilogy/FullStack-Lesson-Plans/02-lesson-plans/01-
html-css-three-days/1-Class-Content/1.3/Activities/1-RelativePaths/RelativePaths/
RelativePaths_WorkingExample/style.css">
```

ALWAYS USE RELATIVE FILE PATHS.

- If you deploy your sites without them, **all of your links will fail**.
  - The same will happen if you move your project from one folder to another.
- Remember, there is no such thing as a "C:" drive on the internet.

© 2016 | Coding Boot Camp - All Rights Reserved

Assignment

1. Unzip the folder sent to you via Slack.

2. Edit the HTML files in all of the “RelativePaths” folders. You need to write relative paths that link the HTML documents with CSS stylesheets.

Tip: Check out the “RelativePaths_WorkingExample” folder.

© 2016 | Coding Boot Camp - All Rights Reserved

CSS Selectors

Box Model

Boxes Upon Boxes

In CSS, every element rests within a series of boxes.
Each box has customizable space properties:
margin, border, and padding.

Typical spacing value: 20px 10px 10px 20px (top, right, bottom, left)

© 2016 | Coding Boot Camp - All Rights Reserved

> YOUR TURN!!

Suggested Time: 10 min

```
#box {

    background-color: #1E5792;
    width: 400px;
    height: 440px;
    margin: 10px 30px 20px 50px;
    color: #fff;
    padding: 25px 10px 30px 20px;
    border-style: solid;
    border-width: 22px;
    border-color: #113152;

}
```

How wide is the blue #box?

How tall is the blue #box?

Total element width = content width + left padding + right padding + left border + right border + left margin + right margin

Total element height = content height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

© 2016 | Coding Boot Camp - All Rights Reserved

> YOUR TURN!!

```
#box {

    background-color: #1E5792;
    width: 400px;
    height: 440px;
    margin: 10px 30px 20px 50px;
    color: #fff;
    padding: 25px 10px 30px 20px;
    border-style: solid;
    border-width: 22px;
    border-color: #113152;

}
```

How wide is the blue #box?

How tall is the blue #box?

Total element width = content width + left padding + right padding + left border + right border + left margin + right margin

Total element height = content height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

Answer

Width: 474 px (no margin), 554 px (with margin)

Height: 539 px (no margin), 569 px (with margin)

© 2016 | Coding Boot Camp - All Rights Reserved

> YOUR TURN!!

Answer

Width: 474 px (no margin), 554 px (with margin)

Height: 539 px (no margin), 569 px (with margin)

© 2016 | Coding Boot Camp - All Rights Reserved

We Be Floatin’

Take a Facebook Break…

Warning!

These next topics are fairly “tricky”… **but VERY IMPORTANT.**

Time to channel that inner genius.

© 2016 | Coding Boot Camp - All Rights Reserved

The Concept of “Flow”

- By default, every HTML element displayed in the browser is governed by a concept called **flow.**

- This means that HTML elements force their adjacent elements to **flow around them**.

© 2016 | Coding Boot Camp - All Rights Reserved

Flow Analogy to MS Word

- This concept of “flow” is very similar to the **wrap-text options** you may be familiar with in Microsoft Word.

- Just as in MS Word, you can have images in-line with text, on-top of text, etc.

© 2016 | Coding Boot Camp - All Rights Reserved

Block Elements

- By default, web clients render many HTML elements as **block elements.** Paragraphs, headers, divs and more receive this treatment.

- A block element will take up an entire line of space—unless you intervene with CSS properties.

© 2016 | Coding Boot Camp - All Rights Reserved

Block Elements vs. Inline Elements

- Now contrast the block elements with **inline elements**.

- By using **float CSS** properties, we can command our website to display multiple HTML elements adjacently.

© 2016 | Coding Boot Camp - All Rights Reserved

Floating

CSS

```
#sidebar {
    float: left;
}

#main-content {
    float: right;
}
```

- To transform these block elements into inline elements, we use a CSS property called **float.**

- Floats are **<u>necessary</u>** for building web layouts.

© 2016 | Coding Boot Camp - All Rights Reserved

Clearing the Float

- **Floats often get in the way of our layouts**.
- Sometimes we don't want to give each element the "inline" treatment.

© 2016 | Coding Boot Camp - All Rights Reserved

Clearfix Hack

- Sometimes when elements don’t match up in size, we get situations like the above…

© 2016 | Coding Boot Camp - All Rights Reserved

Clearfix Hack

```
<div class="clearfix">
  Much better!
```

- We can get around this by using “the clearfix hack.”

© 2016 | Coding Boot Camp - All Rights Reserved

Clearfix Hack

```
.clearfix::after {
    content: "";
    display: block;
    clear: both;
}
```

- **::after** is what we call a pseudo-element. We use it to style specific parts of an element.

- This will add an HTML element, hidden from view, after the content of the **".clearfix"** element. This clears the float.

© 2016 | Coding Boot Camp - All Rights Reserved

Quick Demo!

© 2016 | Coding Boot Camp - All Rights Reserved

Quick Demo!

2000x200

300x400

900x400

500x100

© 2016 | Coding Boot Camp - All Rights Reserved

Quick Demo

Instructor: Demo

(2-FloatExamples)

© 2016 | Coding Boot Camp - All Rights Reserved

Fantastic Guide on Floats ****

CSS-TRICKS

All About Floats

CHRIS COYIER // JULY 8, 2009

What is "Float"?

Float is a CSS positioning property. To understand its purpose and origin, we can look to print design. In a print layout, images may be set into the page such that text wraps around them as needed. This is commonly and appropriately called "text wrap". Here is an example of that.

In page layout programs, the boxes that hold the text can be told to honor the text wrap, or to ignore it. Ignoring the text wrap will allow the words to flow right over the image like it wasn't even there. This is the difference between that image being part of the *flow* of the page (or not). Web design is very similar.

- **To all serious front-end developers (this is a <u>necessary</u> read):** https://css-tricks.com/all-about-floats/

© 2016 | Coding Boot Camp - All Rights Reserved

Assignment

In this activity, you'll flex your newfound floating skills by creating a conceptual layout. Eyeball the design to your best ability.

Check your Slack for more instructions.

© 2016 | Coding Boot Camp - All Rights Reserved

> YOUR TURN!!

© 2016 | Coding Boot Camp - All Rights Reserved

Good work!

Your brain may rest now…

© 2016 | Coding Boot Camp - All Rights Reserved

Video Walkthrough! (Highly, HIGHLY Recommend!!!)

Video Link: https://youtu.be/0lpxKw6E90Y

© 2016 | Coding Boot Camp - All Rights Reserved

BREAK!

CSS Positioning

Position: Static (Default)

- **Four boxes placed statically (default)**

© 2016 | Coding Boot Camp - All Rights Reserved

Position: Relative

- **Switching the boxes to relative will nudge the boxes in relation to their "original" location.**

```
.box {
  background: #2db34a;
  height: 80px;
  position: relative;
  width: 80px;
}
.box-1 {
  top: 20px;
}
.box-2 {
  left: 40px;
}
.box-3 {
  bottom: -10px;
  right: 20px;
}
```

© 2016 | Coding Boot Camp - All Rights Reserved

Position: Absolute

```
.box-set {
  height: 400px;
  background: darkgray;
  position: relative;
}
.box {
  position: absolute;
  height: 150px;
  width: 150px;
  background: #2db34a;
  border: 2px solid black;
}
.box-1 {
  top: 6%;
  left: 2%;
}
.box-2 {
  top: 0;
  right: -40px;
}
.box-3 {
  bottom: -10px;
  right: 20px;
}
.box-4 {
  bottom: 0;
}
```

Positioned relative to nearest positioned ancestor

© 2016 | Coding Boot Camp - All Rights Reserved

Position: Fixed

```
.box-set {
  height: 400px;
  background: darkgray;
}
.box {
  position: fixed;
  height: 150px;
  width: 150px;
  background: #2db34a;
  border: 2px solid black;
}
.box-1 {
  top: 6%;
  left: 2%;
}
.box-2 {
  top: 0;
  right: -40px;
}
.box-3 {
  bottom: -10px;
  right: 20px;
}
.box-4 {
  bottom: 0;
}
```

Position with exact coordinates to the browser window

© 2016 | Coding Boot Camp - All Rights Reserved

Layering with Z-Index

- **Z-Index allows you to layer elements on top of each other when they're positioned.**

© 2016 | Coding Boot Camp - All Rights Reserved

Hiding Things

- “Display: none” allows us to hide elements from view.

- This will become useful in later sections, when we’ll be hiding and revealing specific HTML elements of our choice.

© 2016 | Coding Boot Camp - All Rights Reserved

Quick Demo

Instructor: Demo

(5-CSS_PositionedLayout)

© 2016 | Coding Boot Camp - All Rights Reserved

Video Walkthrough!

Video Link: https://youtu.be/sHfJn0jqBro

© 2016 | Coding Boot Camp - All Rights Reserved

Great Resource

Learn to Code
HTML & CSS

Lesson 1
Building Your First Web Page

Lesson 2
Getting to Know HTML

Lesson 3
Getting to Know CSS New

Lesson 4
Opening the Box Model

Lesson 5
Positioning Content New

Lesson 6

Lesson 5

Positioning Content

One of the best things about CSS is that it gives us the ability to position content and elements on a page in nearly any imaginable way, bringing structure to our designs and helping make content more digestible.

There are a few different types of positioning within CSS, and each has its own application. In this chapter we're going to take a look at a few different use cases—creating reusable layouts and uniquely positioning one-off elements—and describe a few ways to go about each.

Positioning with Floats

One way to position elements on a page is with the `float` property. The `float` property is pretty versatile and can be used in a number of different ways.

Essentially, the `float` property allows us to take an element, remove it from the normal flow of a page, and position it to the left or right of its parent element. All other

In this Lesson

5

CSS

- Positioning with Floats
- Positioning with Inline-Block
- Creating Reusable Layouts
- Uniquely Positioning Elements

SHARE

- **Another great read for front-end developers:** http://learn.shayhowe.com/html-css/positioning-content/

© 2016 | Coding Boot Camp - All Rights Reserved

Assignment

In this activity, you'll flex your newfound positioning skills by creating another conceptual layout. Eyeball the design to your best ability.

Check your Slack for additional instructions.

© 2016 | Coding Boot Camp - All Rights Reserved

> YOUR TURN!!

© 2016 | Coding Boot Camp - All Rights Reserved

Video Walkthrough!

Video Link: https://youtu.be/yWXgnQaWSW0

© 2016 | Coding Boot Camp - All Rights Reserved

Advice

<u>Re-do this at home.</u>

We designed this exercise to firm up your HTML/CSS skills.

<u>REMEMBER:</u>

The best way to learn web development is to PRACTICE!

© 2016 | Coding Boot Camp - All Rights Reserved

Chrome Inspector

Chrome Inspector is Your Friend

- **To access chrome inspector, right click on a page. Then hit "Inspect."**
- It will allow you to inspect the HTML, CSS and more!

© 2016 | Coding Boot Camp - All Rights Reserved

Chrome Inspector is Your Friend

- **You can even edit the HTML/CSS of a webpage and instantly view your changes in the browser!**
- This works on any website, whether yours or not.

© 2016 | Coding Boot Camp - All Rights Reserved

Next Class!

We'll be coming back to this in our next class.

© 2016 | Coding Boot Camp - All Rights Reserved

Recap + Reassurances

You got this

Questions?