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>>>>)

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.

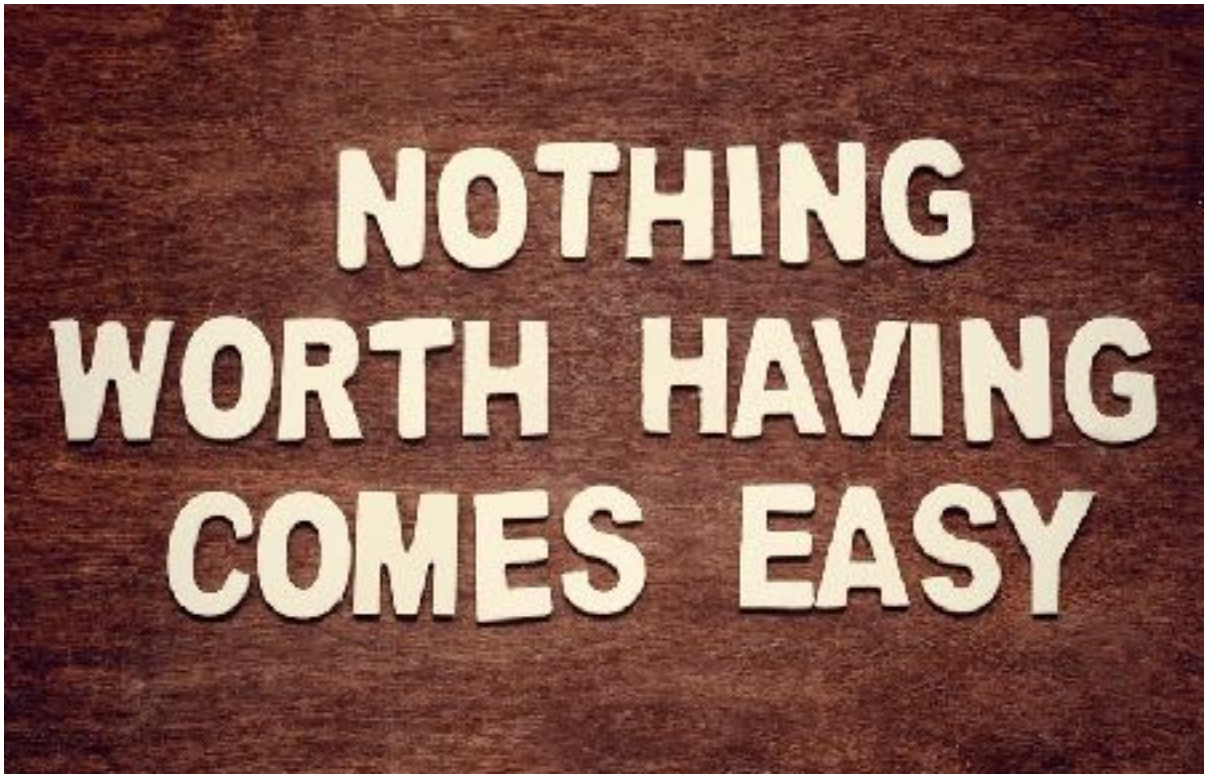
Most Important of All….

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

Warning!

Brace Yourselves

Today is going to be a bit tough.



But trust us!

It will all look easy a few weeks from now.

Brace Yourselves

Don't expect to understand EVERYTHING at once.

Today is all about getting immersed.

CSS Recap

What is “CSS?”

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.)

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.

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

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

Critical Question

How do we style HTML…

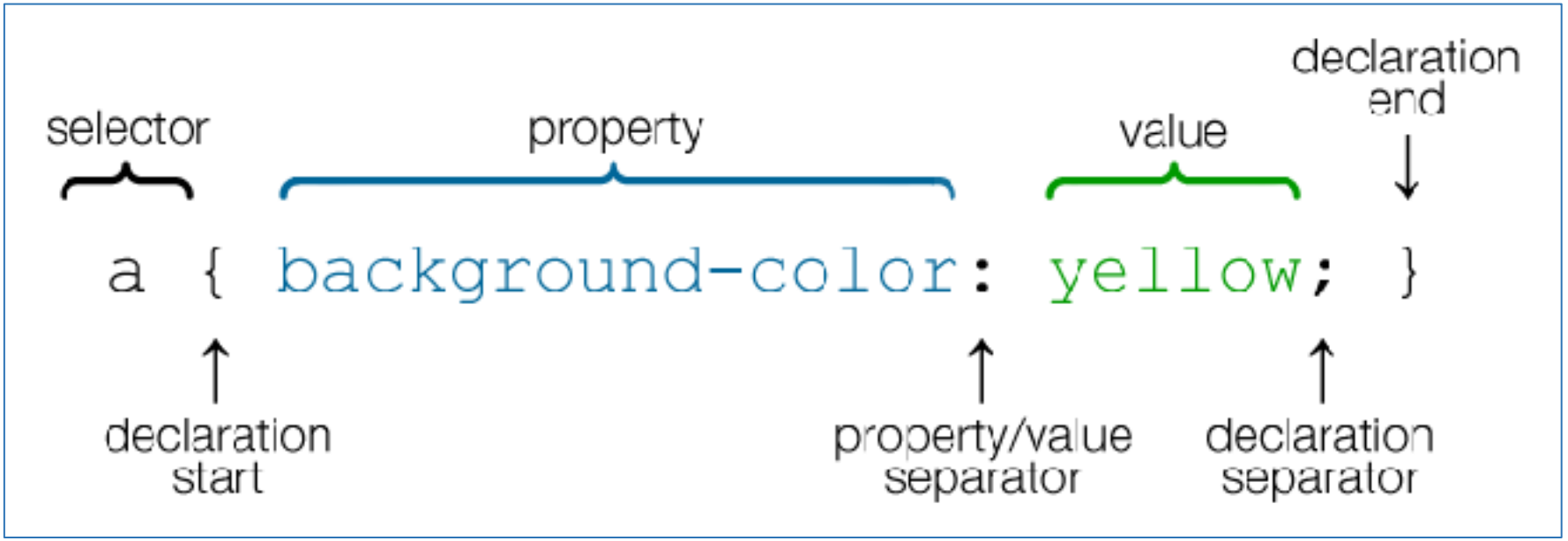
Elements?

Classes?

IDs?

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.

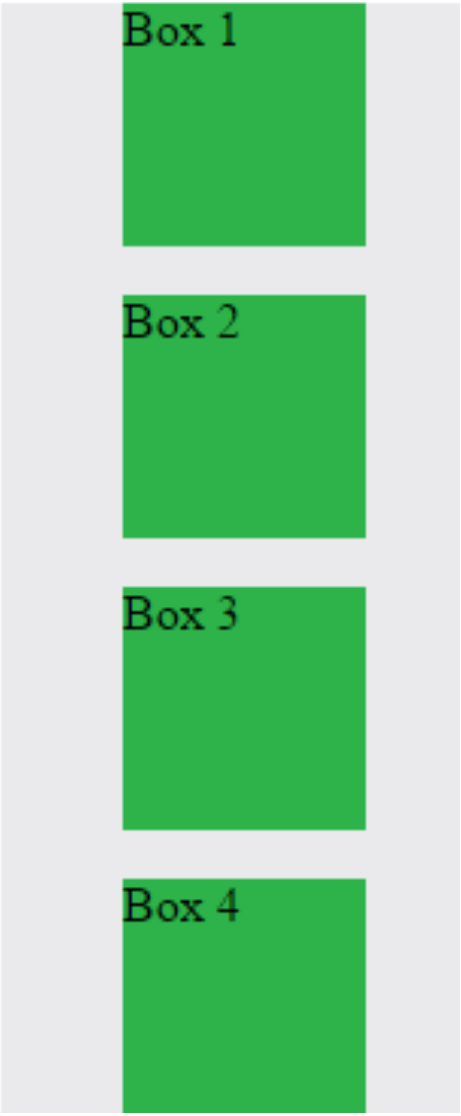


Lingering Questions

Questions so far?

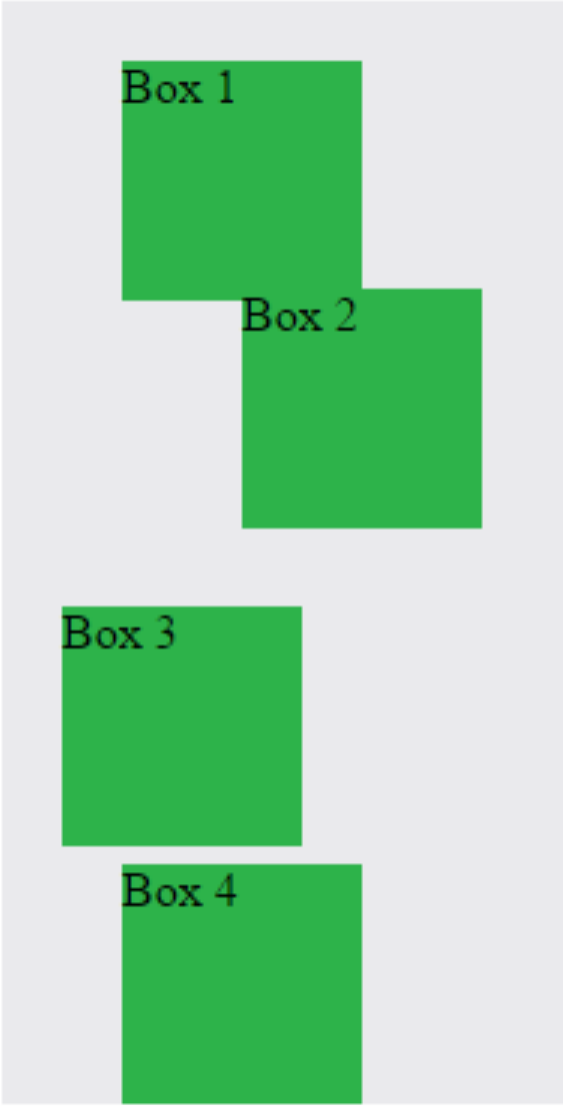
CSS Positioning

Position: Static (Default)



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

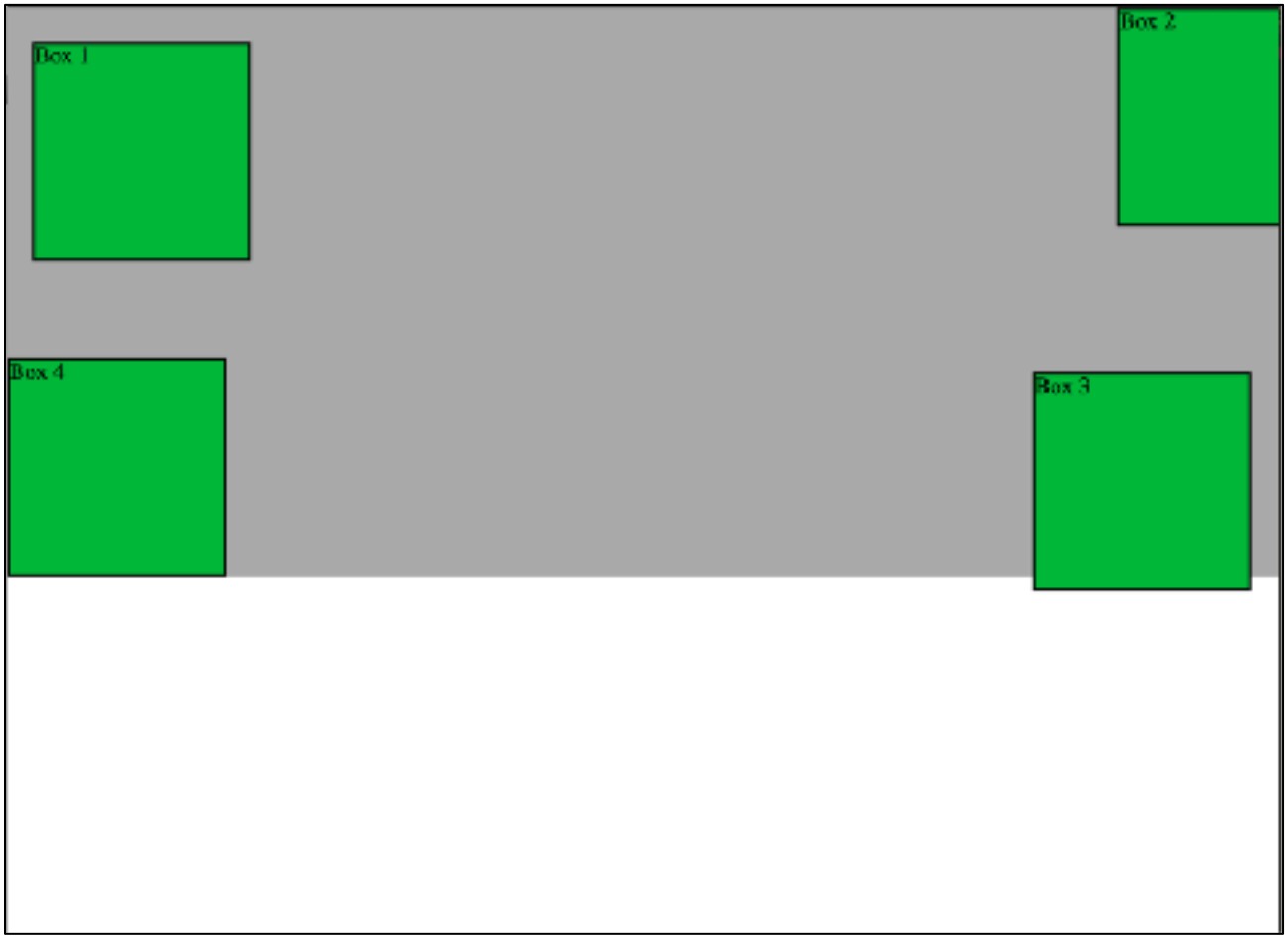
Position: Relative



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

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

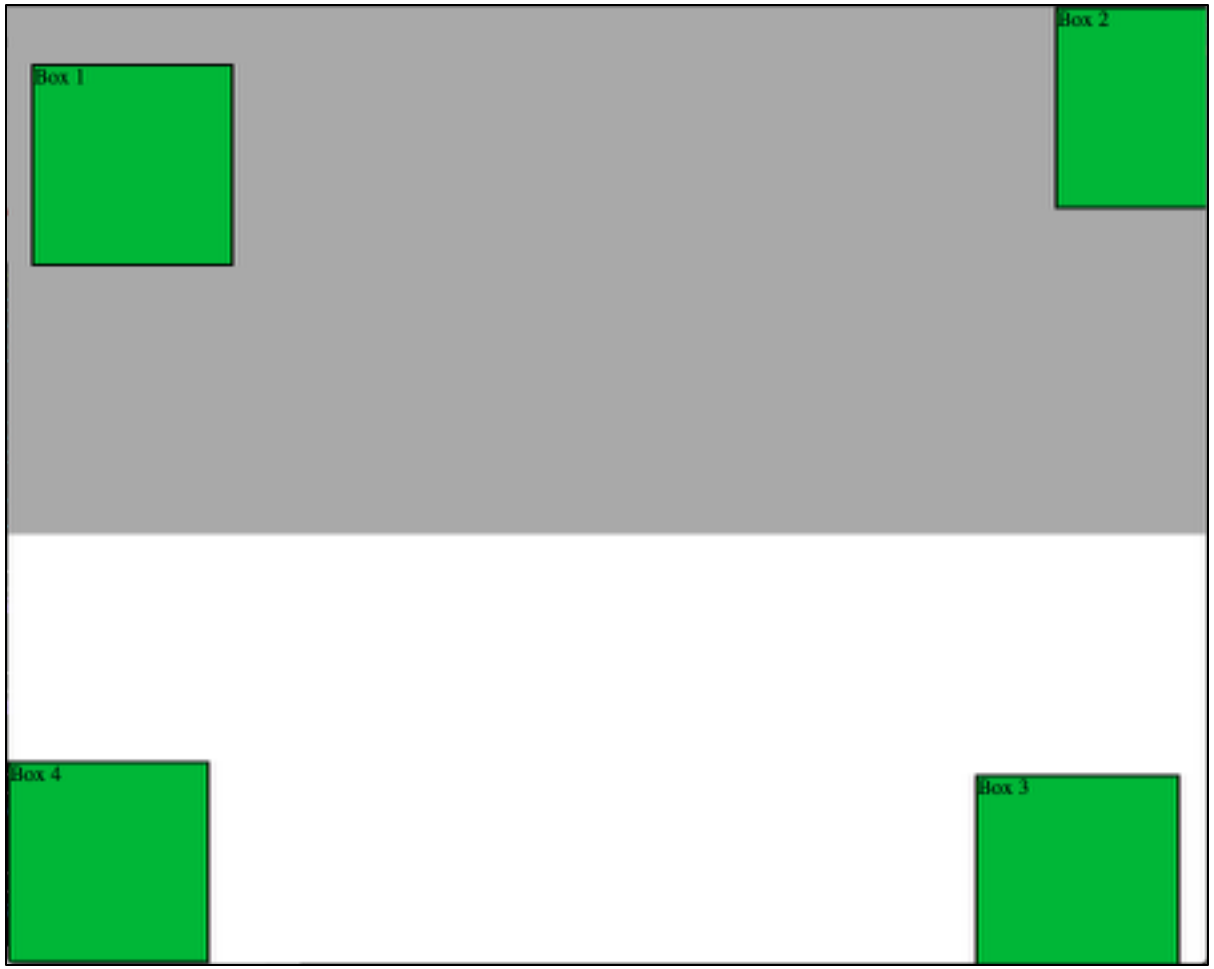
Position: Absolute




```css
.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

Position: Fixed



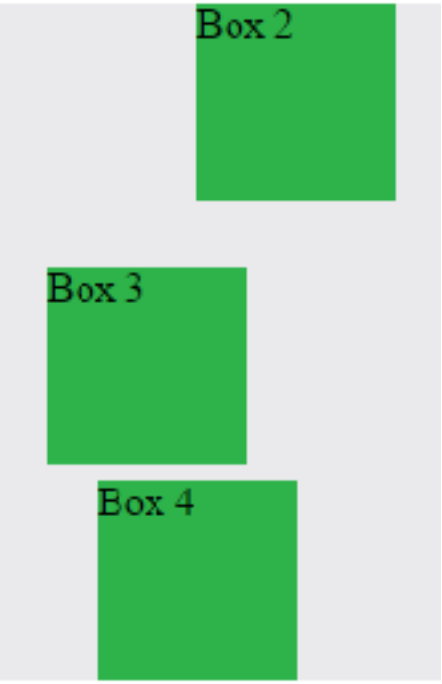

```css
.box-set {
  height: 400px;
  background: darkgray;
}
.box {
  position: fixed;
  height: 150px;
  width: 150px;
  background: #2db34a;
  border: 2px solid black;
}
.box-1 {
  top: 6%;
  left: 2%;
}
.box-2 {
  top: 0;
  right: -40px;
}
.box-3 {
  bottom: -10px;
  right: 20px;
}
.box-4 {
  bottom: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

Position with exact coordinates to the browser window

Layering with Z-Index

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

Hiding Things



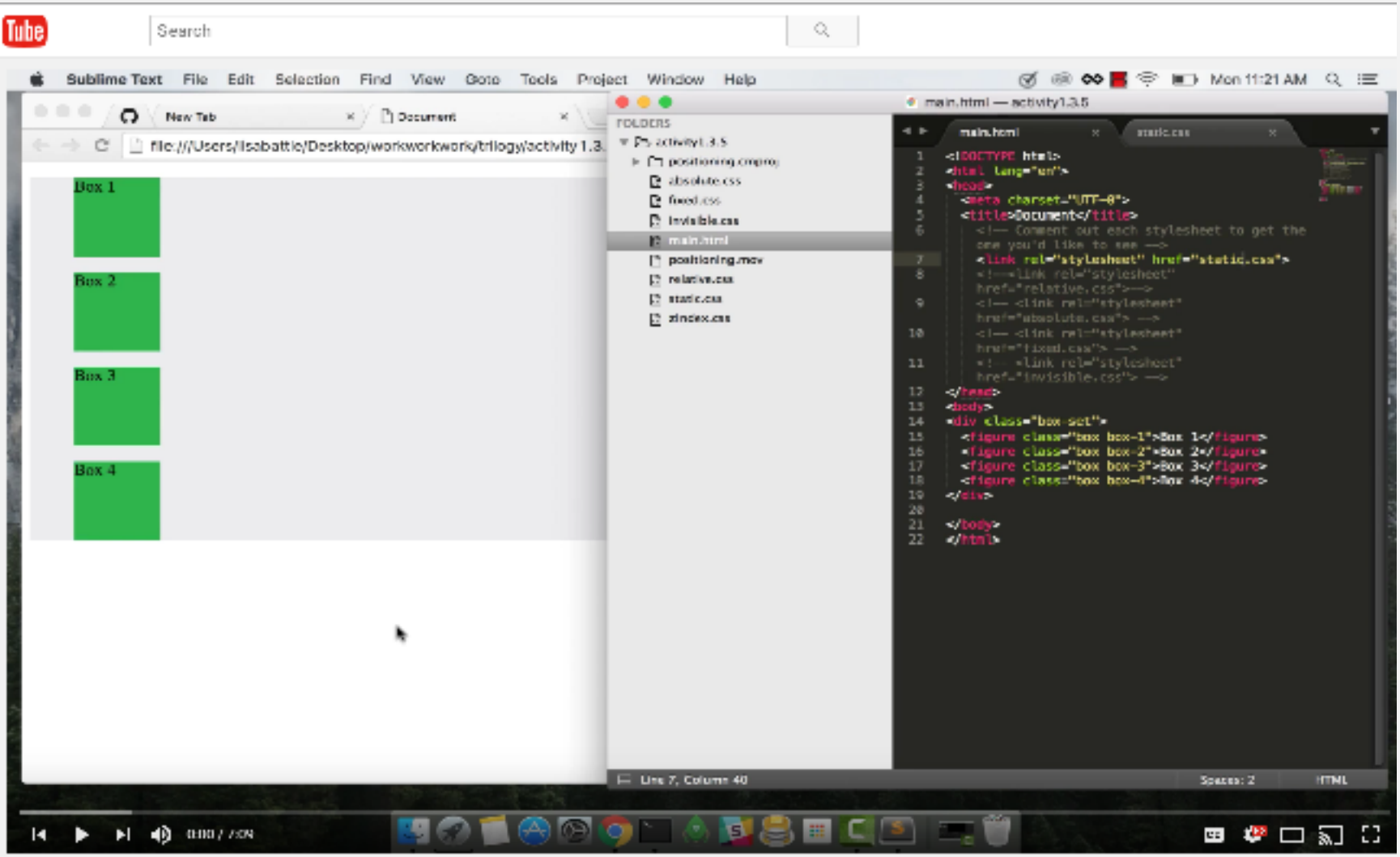
- “Display: none” allows us to hide elements from view.
- This will become useful in later sections, when we’ll be hiding and revealing specific HTML elements of our choice.

Quick Demo

Instructor: Demo

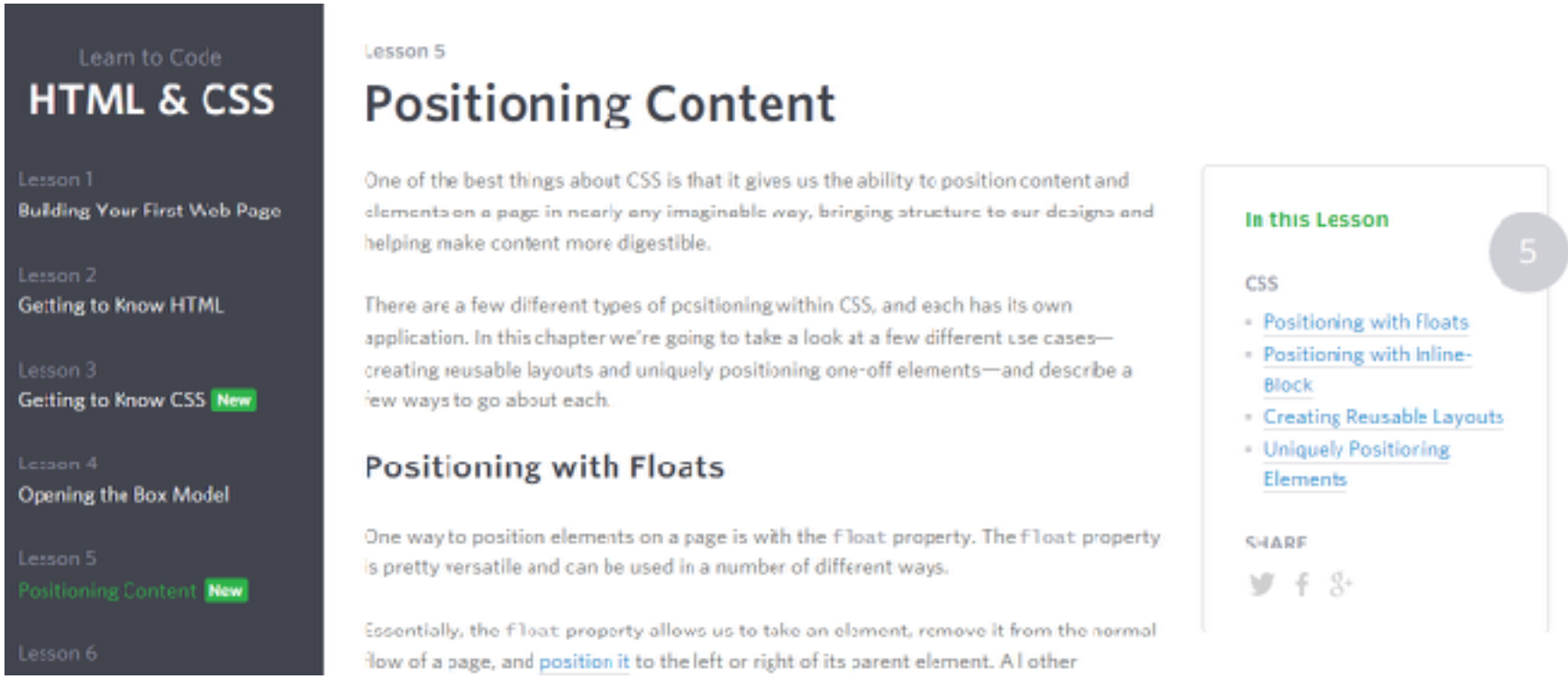
(10-CSS_PositionedLayout)

Video Walkthrough!



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

Great Resource



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

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.

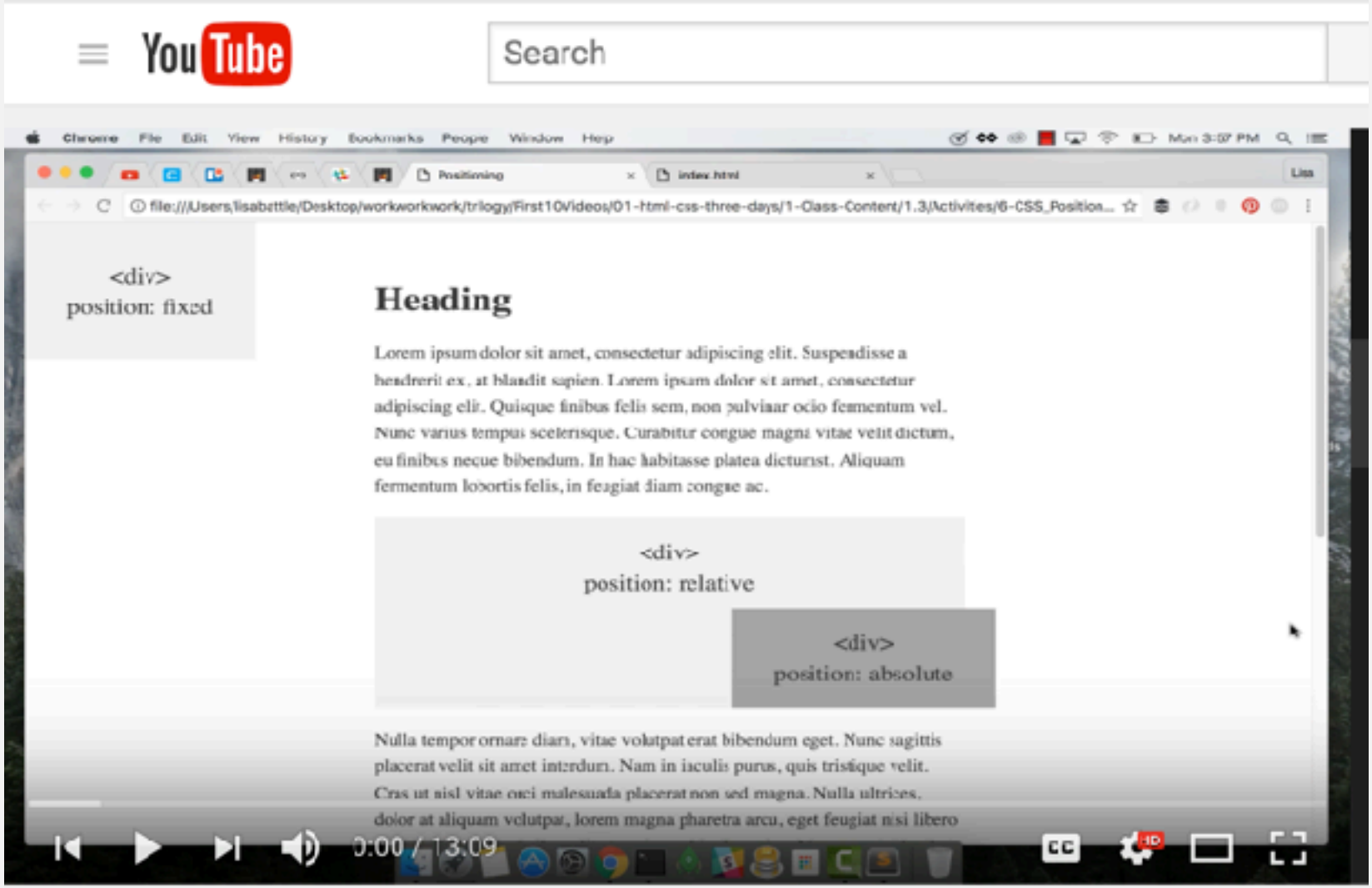
> YOUR TURN!!

position: fixed
ading
ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse a
erit ex, at blandit sapien. Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Quisque finibus felis sem, non pulvinar odio fermentum vel.
Nunc varius tempus scelerisque. Curabitur congue magna vitae velit dictum,
eu finibus neque bibendum. In hac habitasse platea dictumst. Aliquam
fermentum lobortis felis, in feugiat diam congue ac.

position: relative

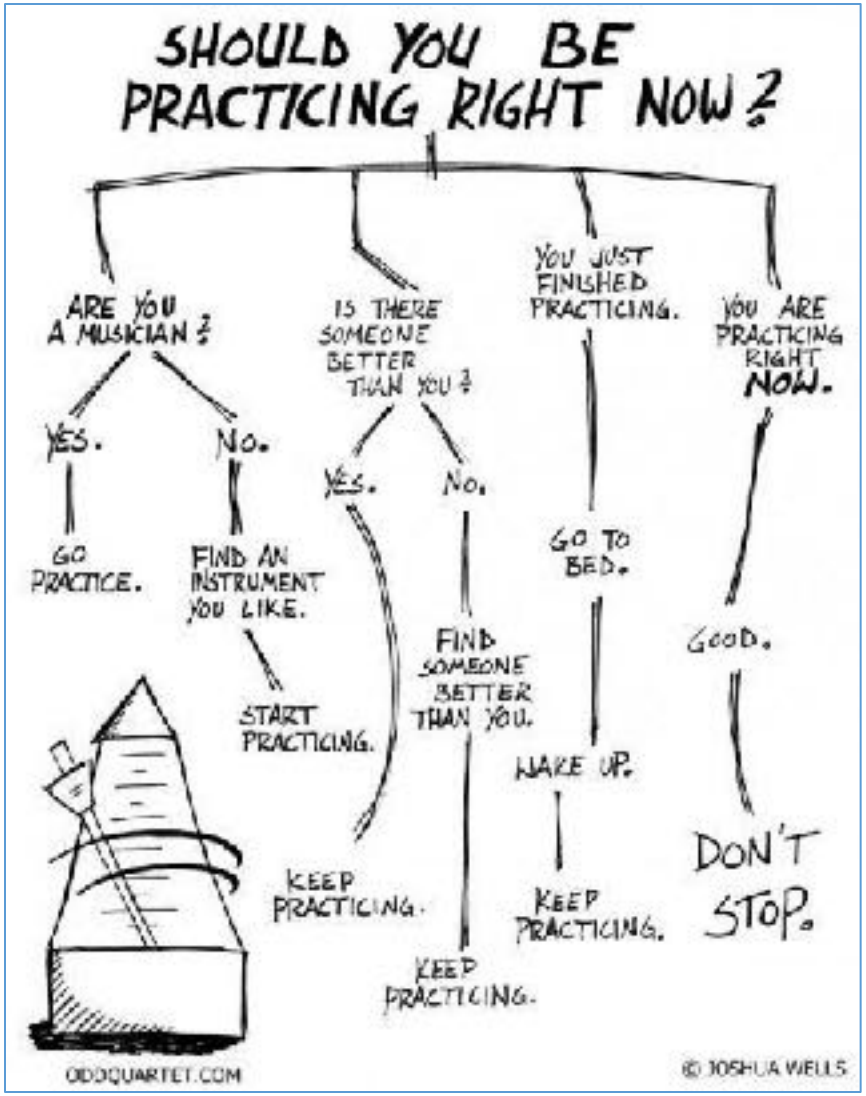
position: absolute
Nulla tempor ornare diam, vitae volutpat erat bibendum eget. Nunc sagittis
placerat velit sit amet interdum. Nam in iaculis purus, quis tristique velit.
Cras ut nisl vitae orci malesuada placerat non sed magna. Nulla ultrices,
dolor at aliquam volutpat, lorem magna pharetra arcu, eget feugiat nisi libero
at nunc. Phasellus finibus elit at sapien vehicula varius. Maecenas in dapibus
leo. Aliquam molestie vulputate metus. Morbi sed posuere quam, et sodales
felis. Proin augue nulla, pellentesque at venenatis vel, sagittis eget nibh.
Maecenas libero velit, luctus eu velit vitae, eleifend convallis felis.

Video Walkthrough!



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

Advice



Re-do this at home.

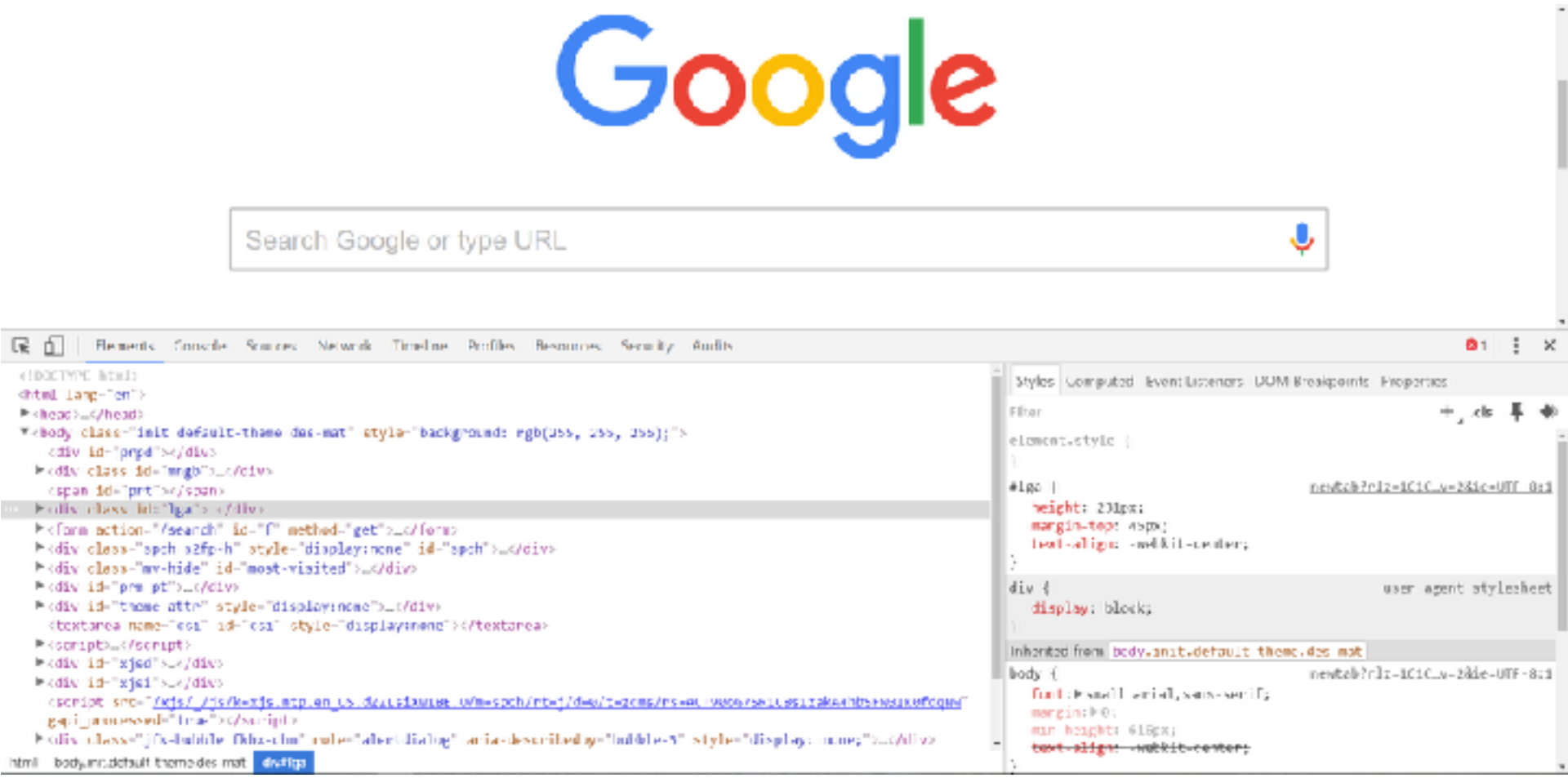
We designed this exercise to firm up your HTML/CSS skills.

REMEMBER:

The best way to learn web development is to PRACTICE!

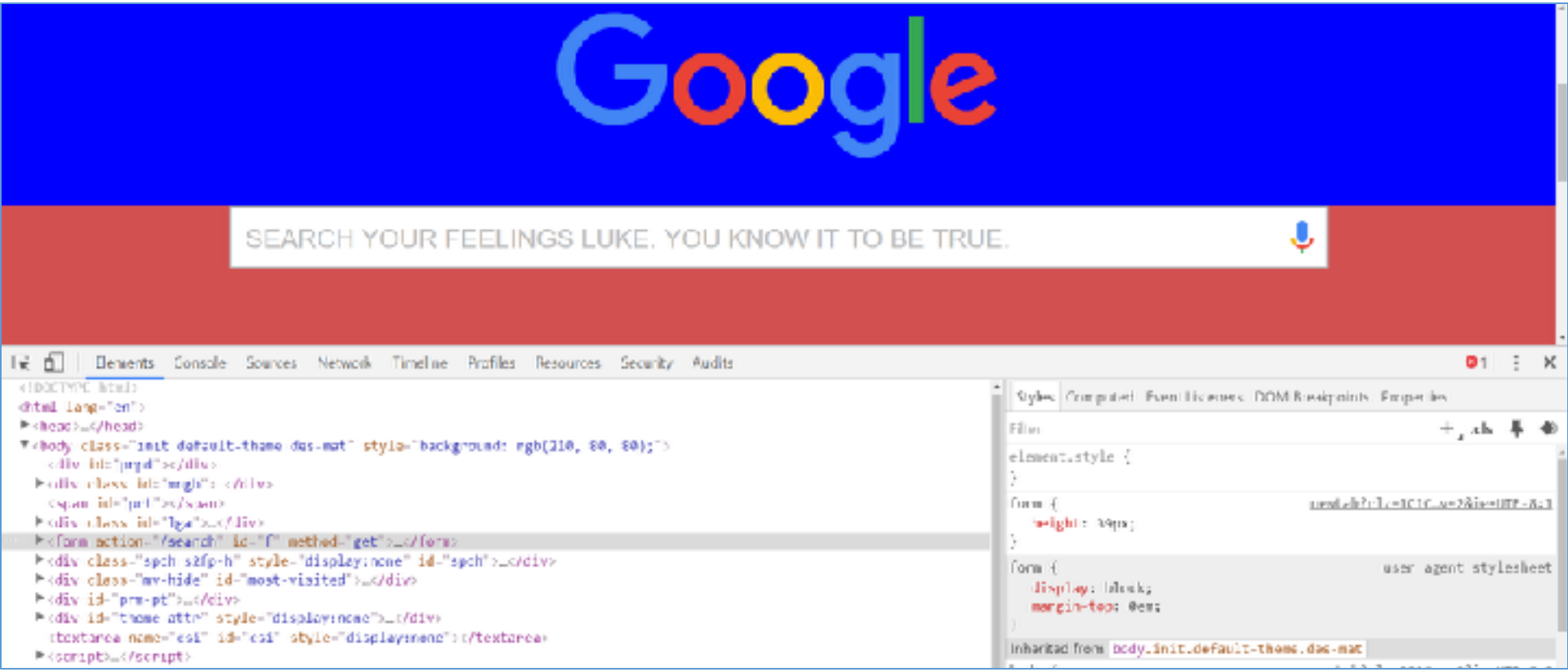
Chrome Inspector

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!

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.

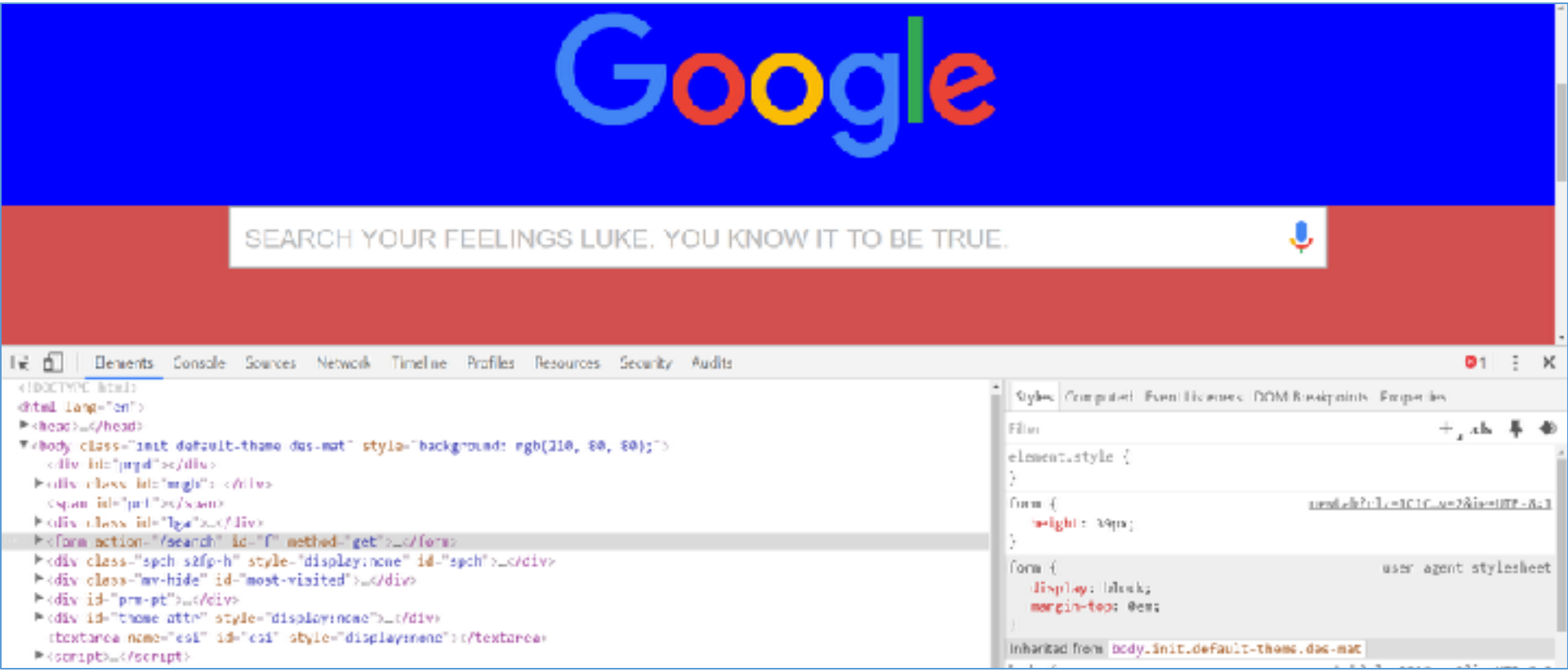
Coming Soon!

We'll be coming back to this after the break.

LUNCH (30 mins)

Chrome Inspector Part Duex

Modifying Sites



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

INSTRUCTOR DEMO!

Instructor: Demo

(Chrome Developer Tools)

> YOUR TURN!!

Suggested Time: 15 min

Assignment

For the next 15 minutes, take a website you commonly use (Amazon, Google, Huff Po, etc.) and heavily modify it using the Chrome Developer Tools.

Be sure to at least modify:

- Content (Change words)
- Colors
- Spacing

Send a screenshot to the class's slack profile when you're done.

> YOUR TURN!!

Suggested Time: 10 min

Assignment

For the next 10 minutes, edit any site that you've been working on in-class or for homework with Chrome Developer Tools.

Be sure to at least modify:

- Content (Change words)
- Colors
- Spacing

Loading Multiple CSS Files
(Very Important!!!)

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Multiple CSS Files!!</title>
5      <link rel="stylesheet" href="assets/style1.css">
6      <link rel="stylesheet" href="assets/style2.css">
7      <link rel="stylesheet" href="assets/style3.css">
8  </head>
9  <body>
10     <header>
```

- An incredibly powerful technique: deploying multiple CSS files simultaneously.

- This lets developers to create complex designs made up of abounding design elements.

- Just remember: ***the loading order matters!!!***

INSTRUCTOR DEMO!

Instructor: Demo

(1-3_CSSFiles.html | 03-MultipleCSS)

What Browser?

By a show of hands…

Which browser do you use?

Battle of the Browsers

- Under the hood, web browsers often **render web pages differently** than their competition.

- These disparities could mean HTML/CSS displaying differently in each web client.

- Because of these potential divergences, web developers need to make their websites **cross-browser compatible**.

Reset.css (or Normalize.css)

- Reset.css will “reset” all browser-specific CSS. This means your site will appear the same in all browsers.

- However, you will have to re-style everything yourself.

INSTRUCTOR DEMO!

Instructor: Demo

(example.html | 04-ResetCSS)

Why CSS Resets Matter

1. It's important for creating browser-compatible websites
2. It's an example of using someone else's CSS in *your* website!!!
3. It's a common Front-End Developer Interview question.

> YOUR TURN!!

Suggested Time: 10 min

Assignment

Follow the instructions given via Slack to incorporate a **reset.css** file into a basic HTML file.

Note the impact the reset file makes after its inclusion.

The Internet

A deep and complex diagram above on how the internet works.

The World Will See Our Greatness!

GitHub provides hosting for static websites – which means we can deploy our websites and applications onto their servers for the world to see.

Together Now…

Let's all login to GitHub

INSTRUCTOR DEMO!

Instructor: Demo

(GitHub Pages Deployment - Personal)

Deploying Static Personal Site – GitHub Pages

Basic Steps:

1. Create a new repo that is named `_username_.github.io`
2. Navigate into a folder and clone the repo into it
3. Build your files
4. Add, commit, and push your changes into the repository

> YOUR TURN!!

Suggested Time: 15 min

Assignment

Time to take your newfangled website and deploy it to the cloud (in this case, GitHub Pages).

Additional instructions to be sent via Slack.

INSTRUCTOR DEMO!

Instructor: Demo

(GitHub Pages Deployment - Project)

Deploying a Static Project Site – GitHub Pages

Basic Steps:

1. Create a new repository on your GitHub account. You can name this repository whatever you would like.

2. Once inside of the repository, create a new file and name it `index.html`

3. Add some very basic HTML into this file, save it, and then navigate into your repository's Settings tab.

4. Scroll down to the GitHub Pages section and then, in the section labeled "Source", select that you would like to use the master branch as your source.

5. Navigate to `<username>.github.io/<repositoryname>` and you will find that your new web page has gone live!

> YOUR TURN!!

Suggested Time: 15 min

Assignment

Build a newfangled website, and deploy it to GitHub Pages as a project instead of a personal site.

Additional instructions to be sent via Slack.

Recap + Reassurances

You got this

Questions?