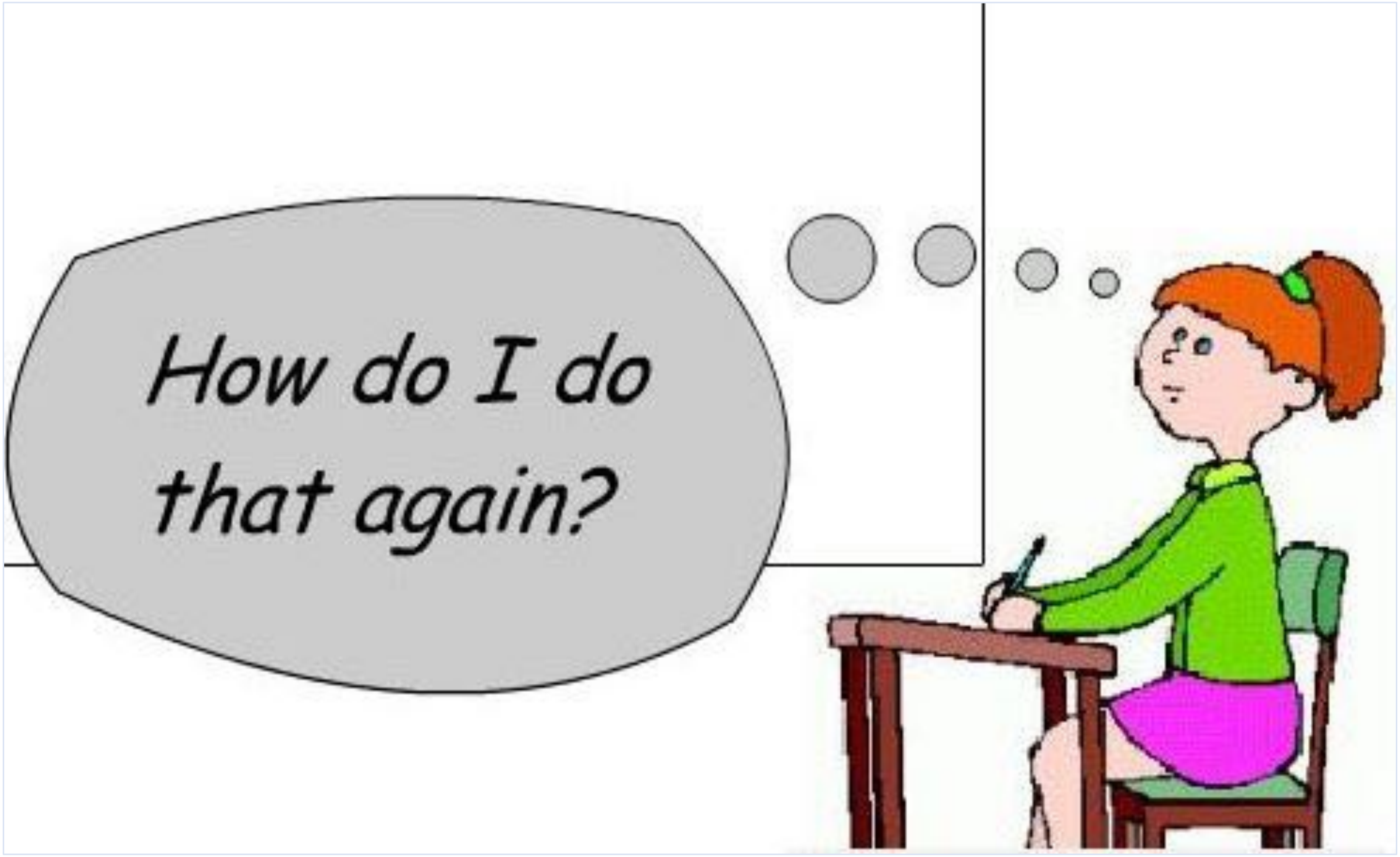
Day 2

Git'n Pro with HTML/CSS

The Coding Bootcamp | October 19, 2017

It’s Okay!



Admin Items

Where to Get Help

- **Practice, Practice, Practice:** Work Individually or in Groups
- **Review In Class Material (Exercises and Slides):** http://ucb.bootcampcontent.com/UCB-Coding-Bootcamp/10-16-2017-UCB-Class-Repository-FSF/tree/master/01-Class-Content/01-html-git-css/01-Activities
- **Re-Watch Class Videos:** https://codingbootcamp.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=2b3e02be-a5cc-4322-ab23-703a307d8dee
- **In Class Office Hours:** 45 minutes before class, 30 minutes after
- **One-on-One Sessions:** By Announcement through SSM
- **Contact Student Success:** Anytime!

Homework #1 - Assignment

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

  http://ucb.bootcampcontent.com/UCB-Coding-Bootcamp/10-16-2017-UCB-Class-Repository-FSF/tree/master/01-Class-Content/01-html-git-css/02-Homework/Instructions

- **Homework Assignment #1 is due next week**
  - **TTH Class: Next Saturday (October 28 2017)**

Today's Class!

Today’s Objectives

- Students will understand the importance of Git Version Control and of how to use it.
- Students will create GitHub Repositories, push code into them, and share with class.
- Students will make more HTML documents.
- Students will learn to properly use basic HTML tags.
- Students will implement basic CSS styling to HTML documents.

Know Thyself

If you are a *complete* beginner to HTML/CSS and Coding:

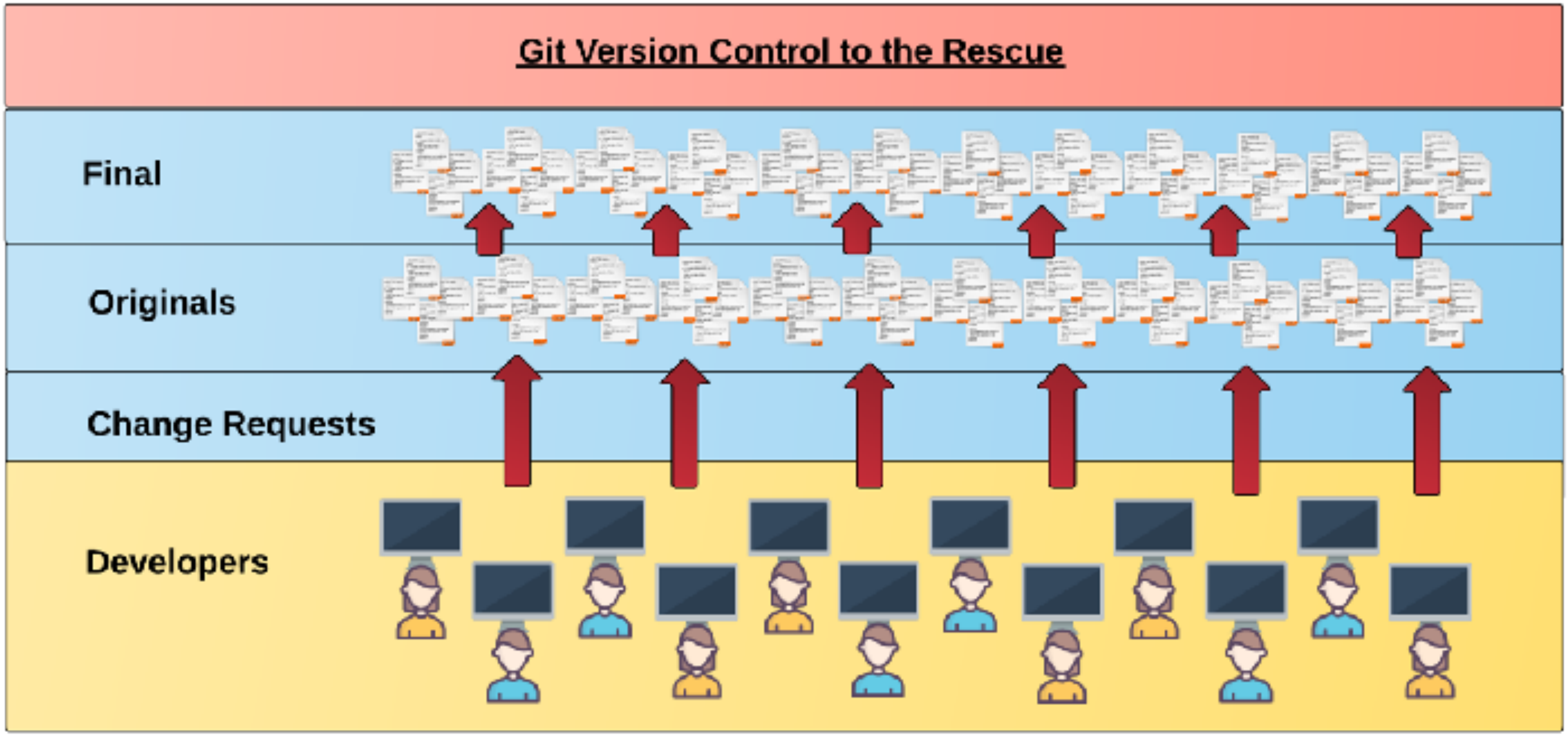
- Continue getting comfortable with HTML.
- Be able to completely write a basic HTML document (like in last class).
- Understand what CSS is, what it's for, and how it works with HTML.
- *Be able to use Git and GitHub to upload code.*

If you've had past exposure and felt comfortable with the last lesson:

- Aim to build up your skills. Clear up any questions or confusions about HTML.
- Become knowledgeable about a wider range of HTML and CSS tags.
- Be able to selectively apply CSS to specific HTML elements.
- *Be able to use Git and GitHub to upload code.*

What / Why Git?

Collaborative Coding



- Modern web development is <u>*highly*</u> collaborative.
- Teams are often extremely large and separated across the country — or planet.
- Apps sometimes comprise hundreds or even thousands of files.

The Team's Task - No Version Control

Task: Make a list in HTML showing the three branches of US government.

Programming Team:

SpongeBob & Kobe make their edits

Programming Away…

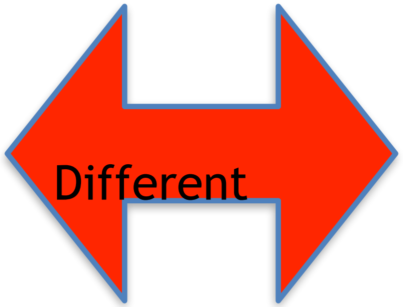
SpongeBob's Version

Programming Away…

Kobe’s Version

Different Solutions

```
<ul>
    <li>Legislative</li>
    <li>Supreme Court</li>
    <li>Executive</li>
</ul>
```



```
<ul>
    <li>Executive</li>
    <li>Congress & Senate</li>
    <li>Judicial</li>
</ul>
```

Resolution

```
<ul>
    <li>Legislative</li>
    <li>Supreme Court</li>
    <li>Executive</li>
</ul>
```

"Let's settle on this…"

```
<ul>
    <li>Legislative</li>
    <li>Judicial</li>
    <li>Executive</li>
</ul>
```

```
<ul>
    <li>Executive</li>
    <li>Congress & Senate</li>
    <li>Judicial</li>
</ul>
```

Kiss dude

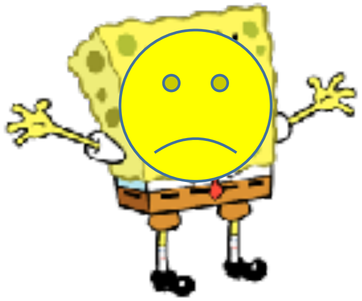
Hai guyz!!!
How R Kan help??

Delete. Delete. Delete. Delete. Delete. Delete

```
~~<ul>~~
  ~~<li>Legislative</li>~~
  ~~<li>Judicial</li>~~
  ~~<li>Executive</li>~~
~~</ul>~~
```

```
<list>
   <li>Washington
   <li>Dudes in Robes/li>
   <li>Mr. Hot Shot<li>
</list>
```

The Group Project



<u>Lesson:</u>

You should use Version Control.

….and watch your teammates' work

Version Control

Git Version Control:

Provides a organized system for managing code for when multiple developers work on a project at the same time.

The Benefits of Git:

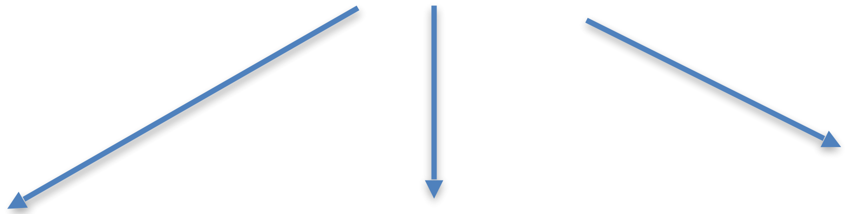
1. A process for resolving conflicts in code.
2. Version History.

The Group Project

Master Branch

'Branch' = personal copy

Personal branch



Kobe's branch

Sponge Bob's branch

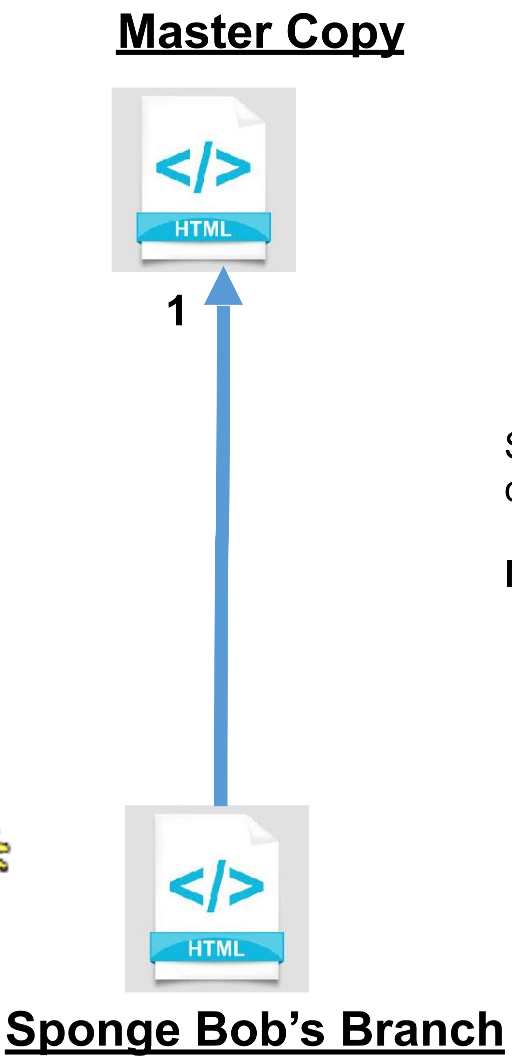
Kiss dude's branch

The team goes to work

```
<ul>
   <li>Legislative</li>
   <li>Supreme Court</li>
   <li>Executive</li>
</ul>
```

```
<ul>
   <li>Executive</li>
   <li>Congress & Senate</li>
   <li>Judicial</li>
</ul>
```

Sponge Bob pushes first



Sponge Bob **pushes (uploads)** his code changes into the main branch.

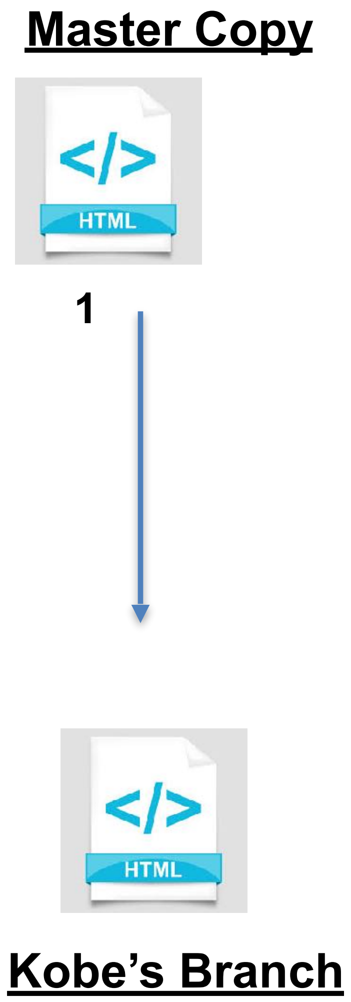
No code conflicts.

Kobe’s edits are ready

Rule: pull first, then push your changes

Ok

Kobe pulls latest changes



Kobe conflicts with master branch

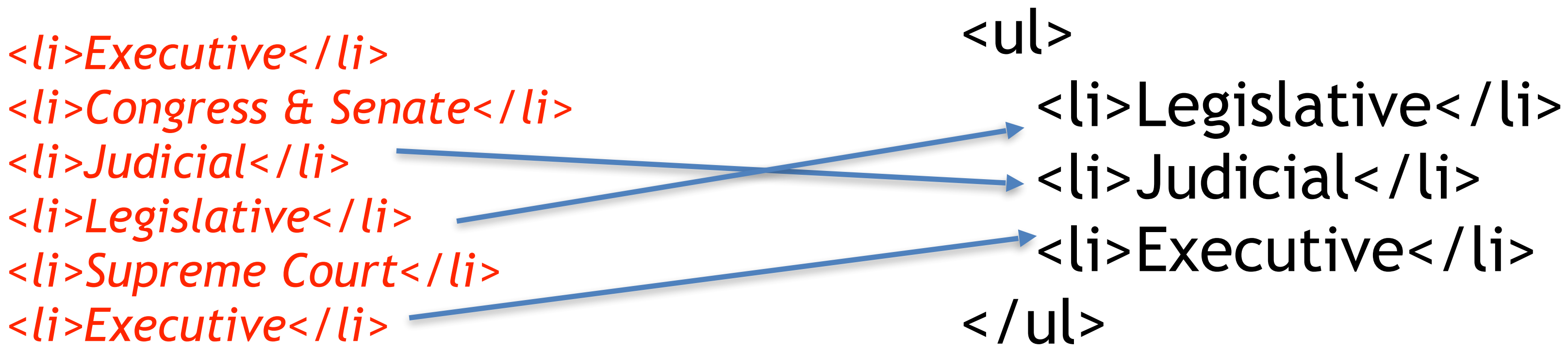
Master Branch

1

```
<li>Executive</li>
<li>Congress & Senate</li>
<li>Judicial</li>
<li>Legislative</li>
<li>Supreme Court</li>
<li>Executive</li>
```

Git sees a conflict.

Kobe resolves



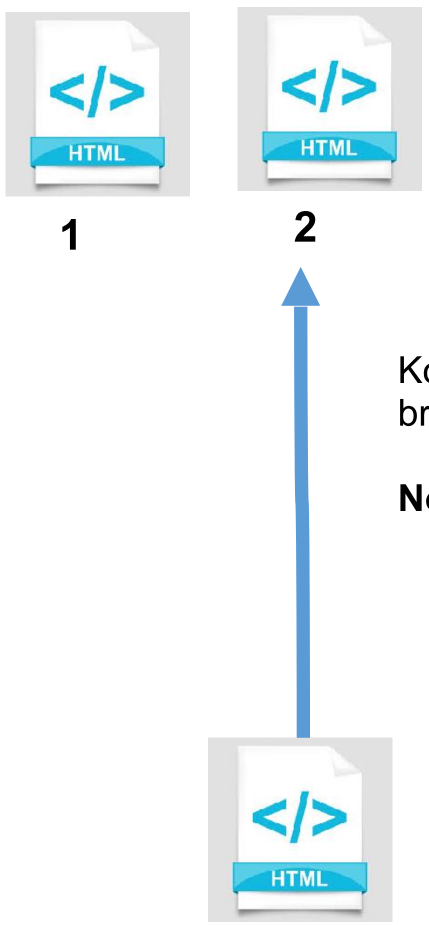
```
<li>Executive</li>
<li>Congress & Senate</li>
<li>Judicial</li>
<li>Legislative</li>
<li>Supreme Court</li>
<li>Executive</li>
```

```
<ul>
   <li>Legislative</li>
   <li>Judicial</li>
   <li>Executive</li>
</ul>
```

Kobe's Branch

Kobe fixes and pushes

Master Branch



Kobe **pushes (uploads)** his revision the main branch.

No code conflicts.

Kobe's Branch

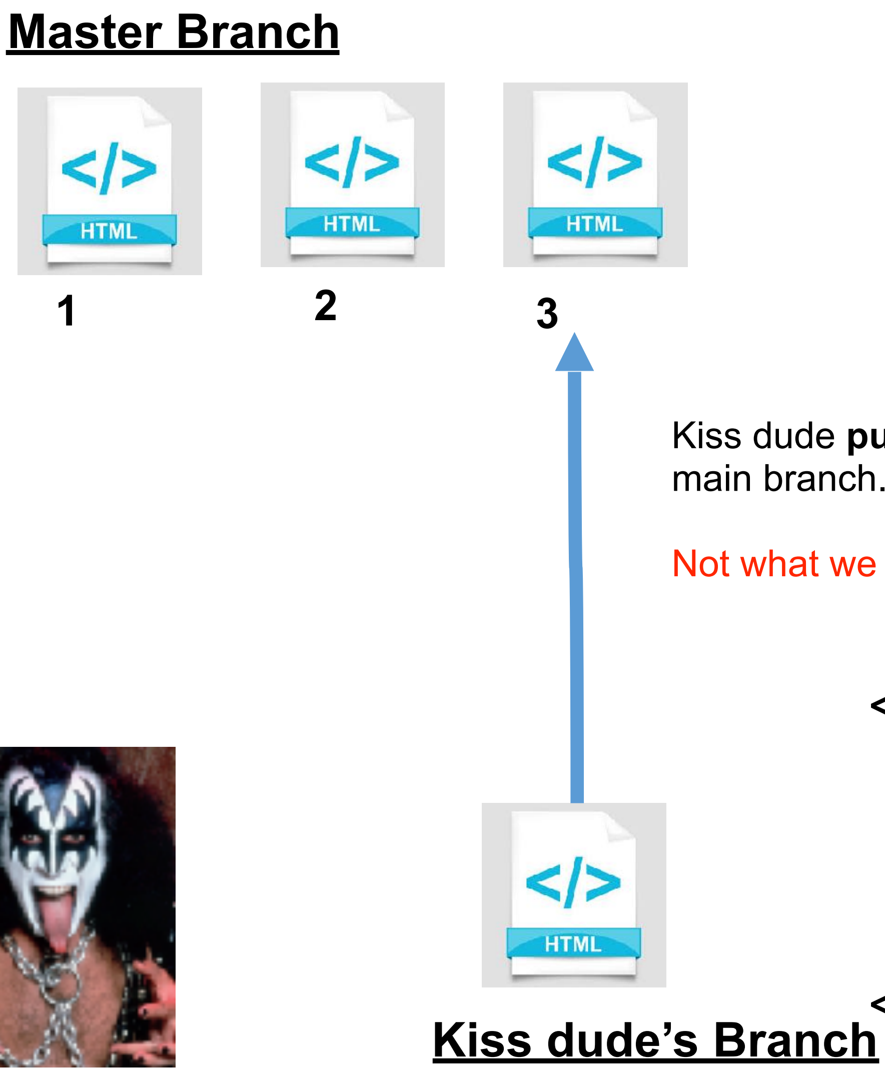
```
<ul>
    <li>Legislative</li>
    <li>Judicial</li>
    <li>Executive</li>
</ul>
```

Kiss Dude starts his work

Rule: pull first, then push your changes

Rules R 4 suckers!!!

Kiss dude pushes



Kiss dude **pushes (uploads)** his revision the main branch. No code conflicts.

Not what we want.

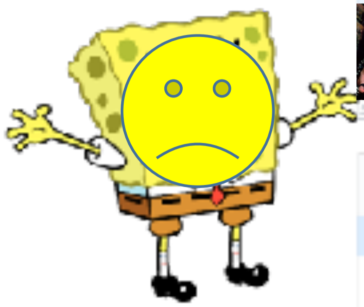
```
<list>
    <li>Washington
    <li>Dudes in Robes/li>
    <li>Mr. Hot Shot<li>
</list>
```

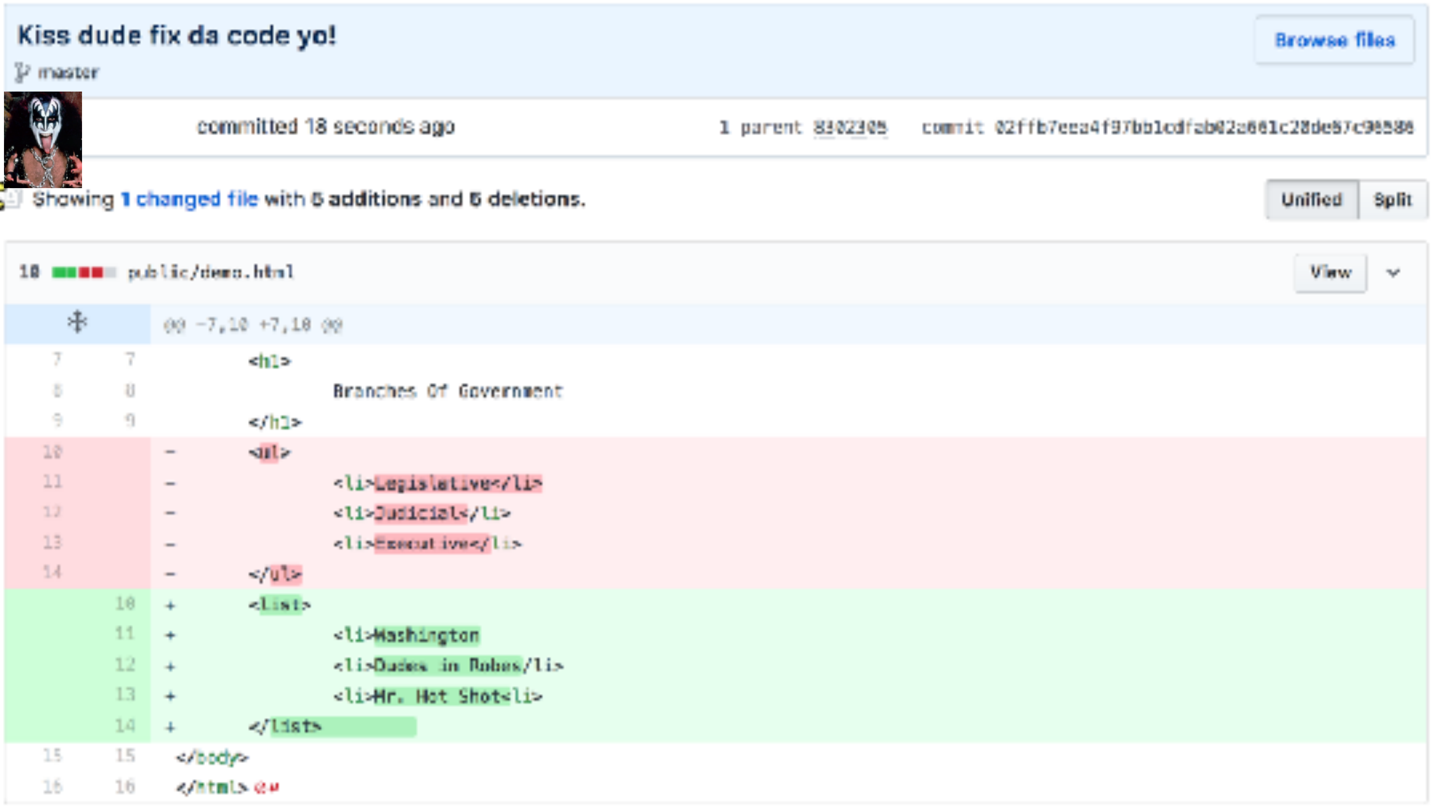
If Kiss dude had made a pull first…

Conflict!

```
<list>
    <li>Washington
    <li>Dudes in Robes/li>
    <li>Mr. Hot Shot<li>
</list>
<ul>
    <li>Legislative</li>
    <li>Judicial</li>
    <li>Executive</li>
</ul>
```

The overwritten work is discovered





Kiss dude fix da code yo!

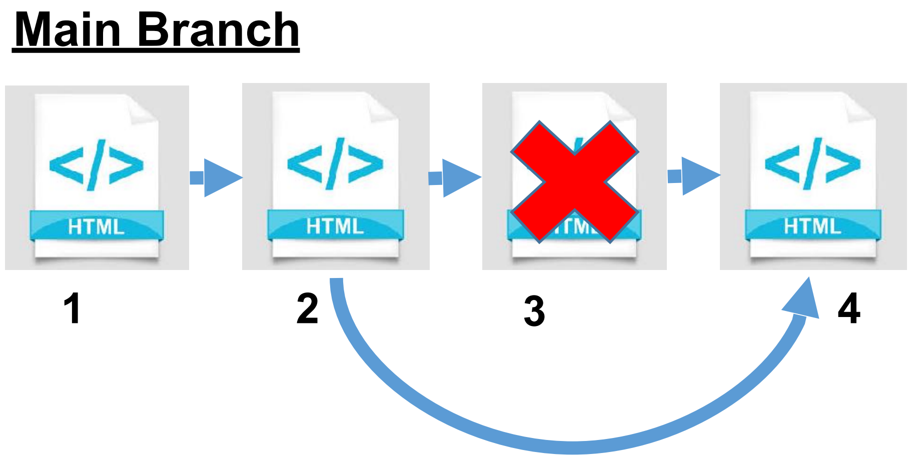
master

committed 18 seconds ago   1 parent 8302305   commit 02ffb7eea4f97bb1cdfab02a661c28de57c98588

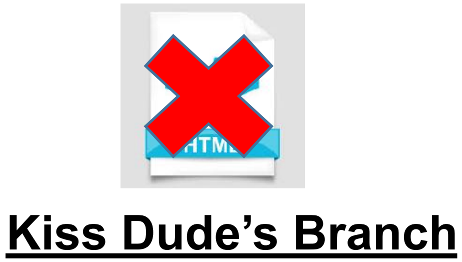
Showing 1 changed file with 5 additions and 5 deletions.

```
10 ■■■■■ public/demo.html

@@ -7,10 +7,10 @@
          <h1>
                Branches Of Government
          </h1>
-         <ul>
-               <li>Legislative</li>
-               <li>Judicial</li>
-               <li>Executive</li>
-         </ul>
+         <list>
+               <li>Washington
+               <li>Dudes in Robes/li>
+               <li>Mr. Hot Shot<li>
+         </list>
      </body>
      </html>
```

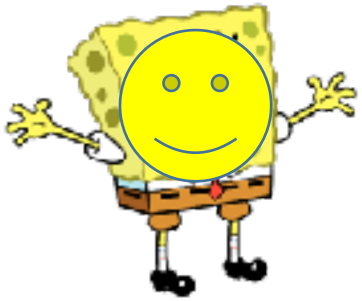
Roll Back



SpongeBob **rolls back** the code to an earlier version.



The Group Project



<u>Lesson:</u>

You should use Version Control!

Quick Activity!

Suggested Time: 3 min

Turn to your neighbor, and have one of you explain to the other:

- **The concept of version control.**

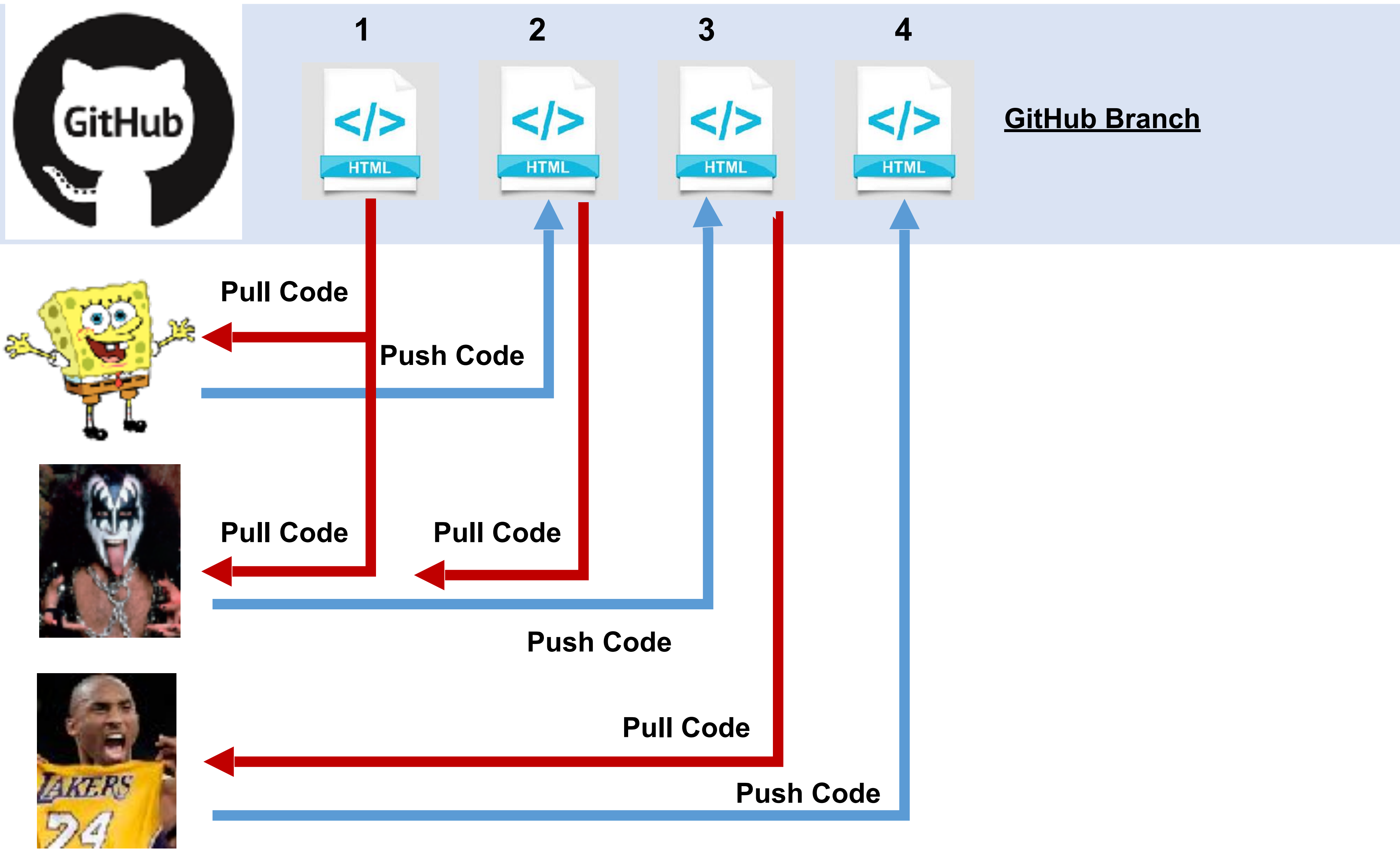
Then the other should explain:

- **Two of the key advantages to using a version control system.**

So… What's this GitHub?

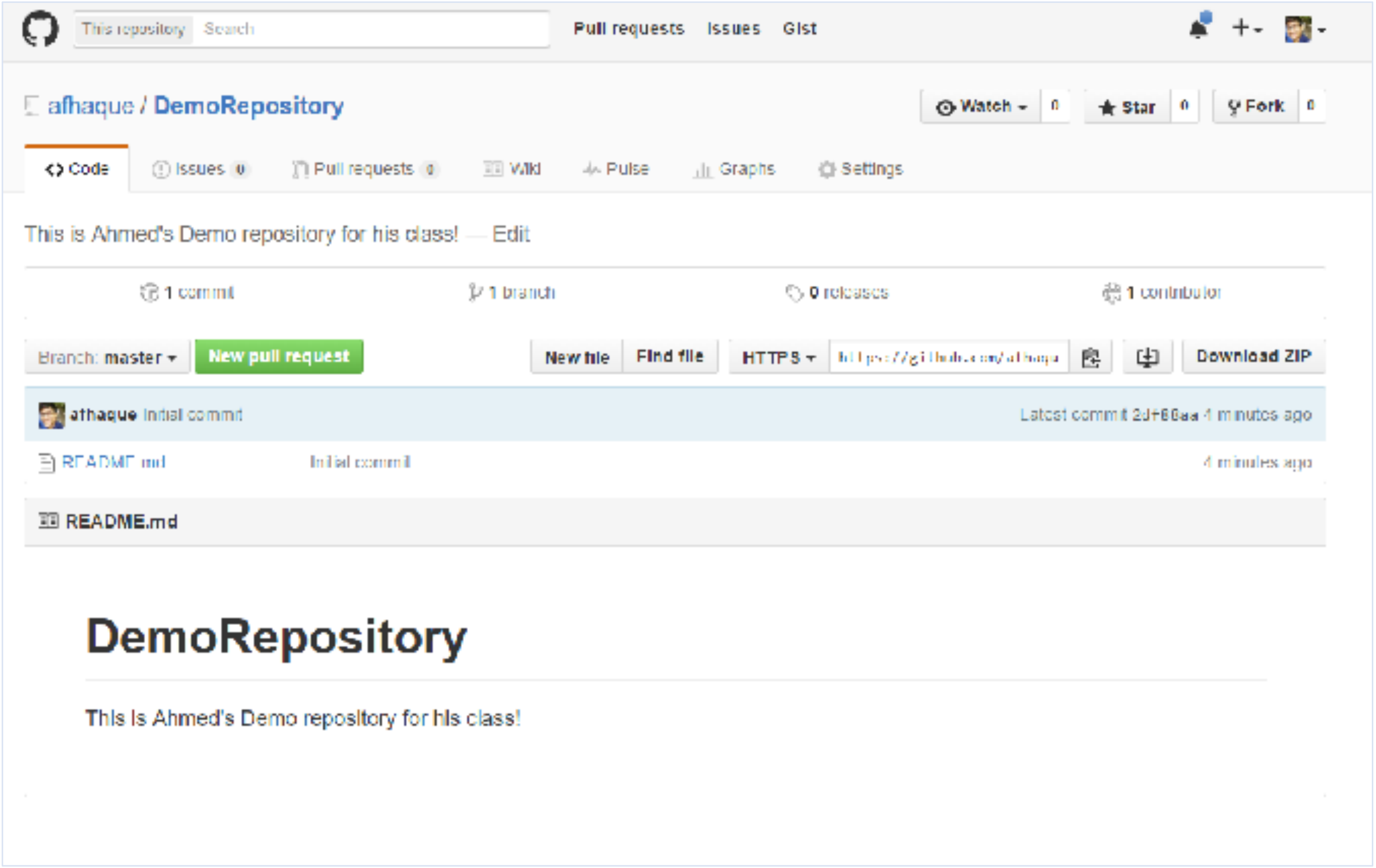
- GitHub is a Web-Based hosting service to store code online.
- It allows developers to **pull** (download) code or **push** (upload) code to the same **repository** (directory).
- It also allows developers to **view histories** of code changes and to **track issues**.

Pushing and Pulling to GitHub



Get Started with Git

Instructor Git Demo!



This repository Search | Pull requests | Issues | Gist

afhaque / DemoRepository

Watch 0 | Star 0 | Fork 0

Code | Issues 0 | Pull requests 0 | Wiki | Pulse | Graphs | Settings

This is Ahmed's Demo repository for his class! — Edit

1 commit | 1 branch | 0 releases | 1 contributor

Branch: master | New pull request | New file | Find file | HTTPS | Download ZIP

afhaque Initial commit — Latest commit 2d+66aa 4 minutes ago

README.md — Initial commit — 4 minutes ago

README.md

DemoRepository

This Is Ahmed's Demo repository for his class!

Basic Git Commands

<u>At its most basic, these are the five git commands to get started:</u>

1. **git clone**
2. **git add**
3. **git commit**
4. **git push**
5. **git pull**

Basic Git Commands

At its most basic, these are the five git commands to get started:

1. **git clone** – copies an entire repo (to begin).
2. **git add** – adds a file for inclusion in Git.
3. **git commit** – notes a change to the local repo.
4. **git push** – sends changes to hosting service.
5. **git pull** – downloads freshest version of repo.

Basic Git Commands

The most useful git command of all…

Basic Git Commands

The most useful git command of all…

```
$ git status
```

Basic Git Commands

The most useful git command of all…

$ git status

When you need a gut check run `git status`

Let Git tell you the current status of your repository. For example, Git will indicate which files have been changed or added as a result of the changes you've made on your branch. Then Git offers suggestions on what to do, offering commands on how to stage or commit those files.

When in doubt, run `git status` and read what Git tells you!

> YOUR TURN!

Activity*:* Git Add, Commit, Push **|** **Suggested Time:** 20 min

Assignment:
Using GitHub and the Command Line:

- Create a new **public GitHub repository** and name it whatever you like. Be sure to check the box for "initialize this repository with a README."

- Next, **clone** the repo to your local directory.

- Then create an HTML file inside the local directory.

- **Add**, **Commit**, and **Push** the code to GitHub.

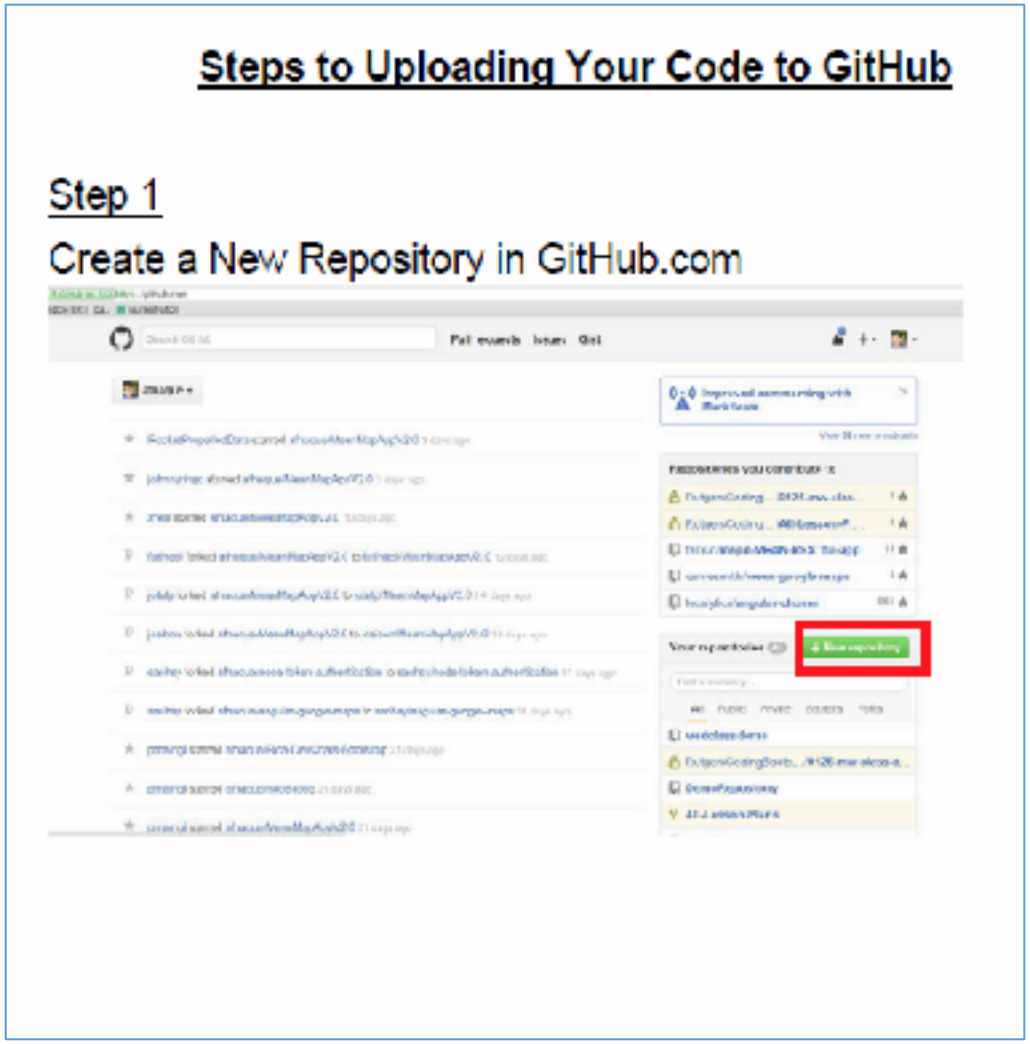
Bonus:

- Find a partner in class, and **fork** *their* repository to your own GitHub account. Clone this forked repository to your local directory.

- Add, Commit, and Push the code back to your forked copy.

- Finally, submit a **pull request** to send your changes to your partner's repo.

Still a Bit Lost? Never Worry!



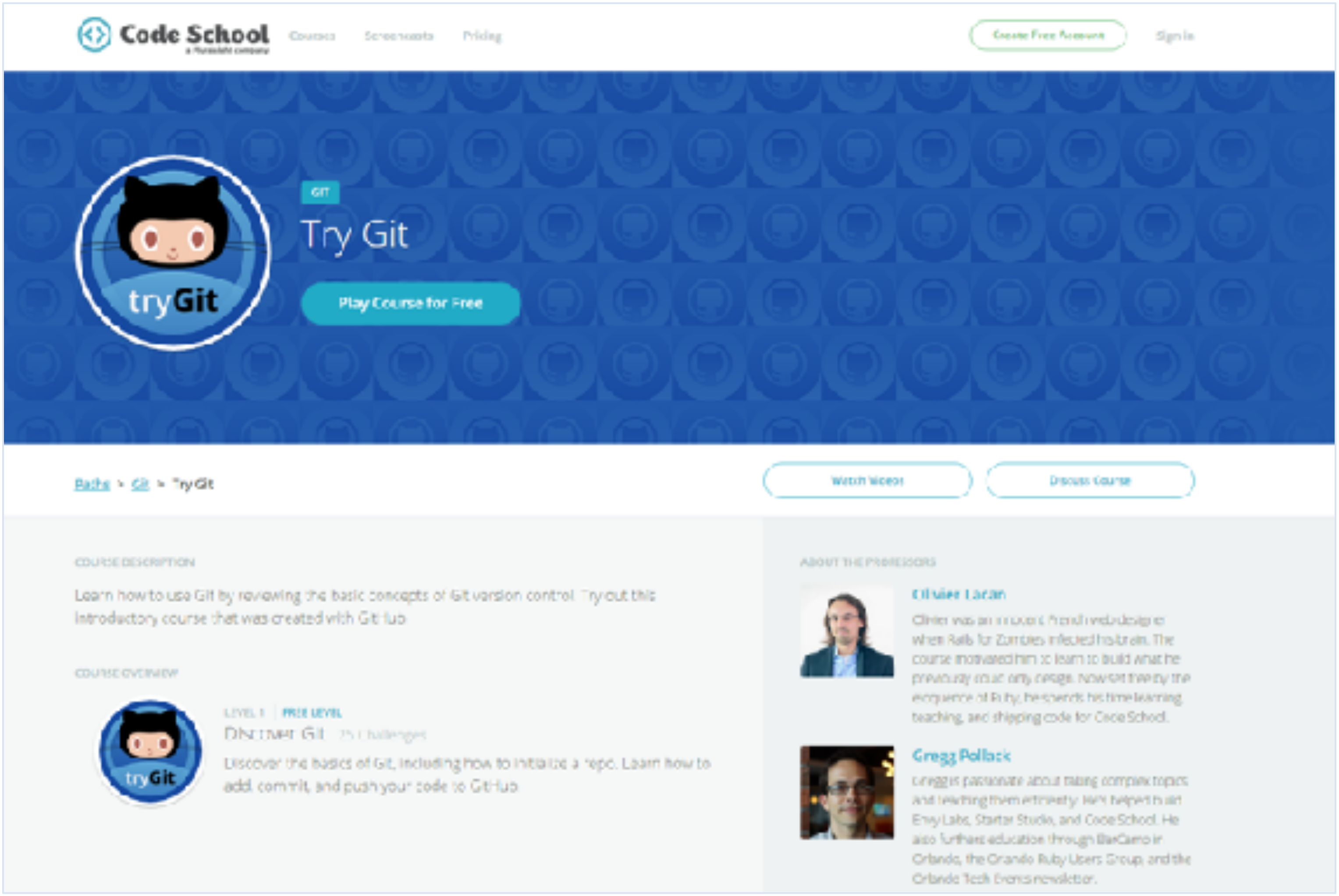
Steps to Uploading Your Code to GitHub

Step 1

Create a New Repository in GitHub.com

- **Follow this handy Guide!**
- Practice a few times on your own before our next class.

If You're Still Lost… Here's a (Free) Course



Try Git

Paths › Git › Try Git

COURSE DESCRIPTION

Learn how to use Git by reviewing the basic concepts of Git version control. Try out this introductory course that was created with GitHub.

COURSE OVERVIEW

LEVEL 1 | FREE LEVEL

Discover Git 25 Challenges

Discover the basics of Git, including how to initialize a repo. Learn how to add, commit, and push your code to GitHub.

ABOUT THE PROFESSORS

Olivier Lacan

Olivier was an innocent French web designer when Rails for Zombies infected his brain. The course motivated him to learn to build what he previously could only design. Now smitten by the elegance of Ruby, he spends his time learning, teaching, and shipping code for Code School.

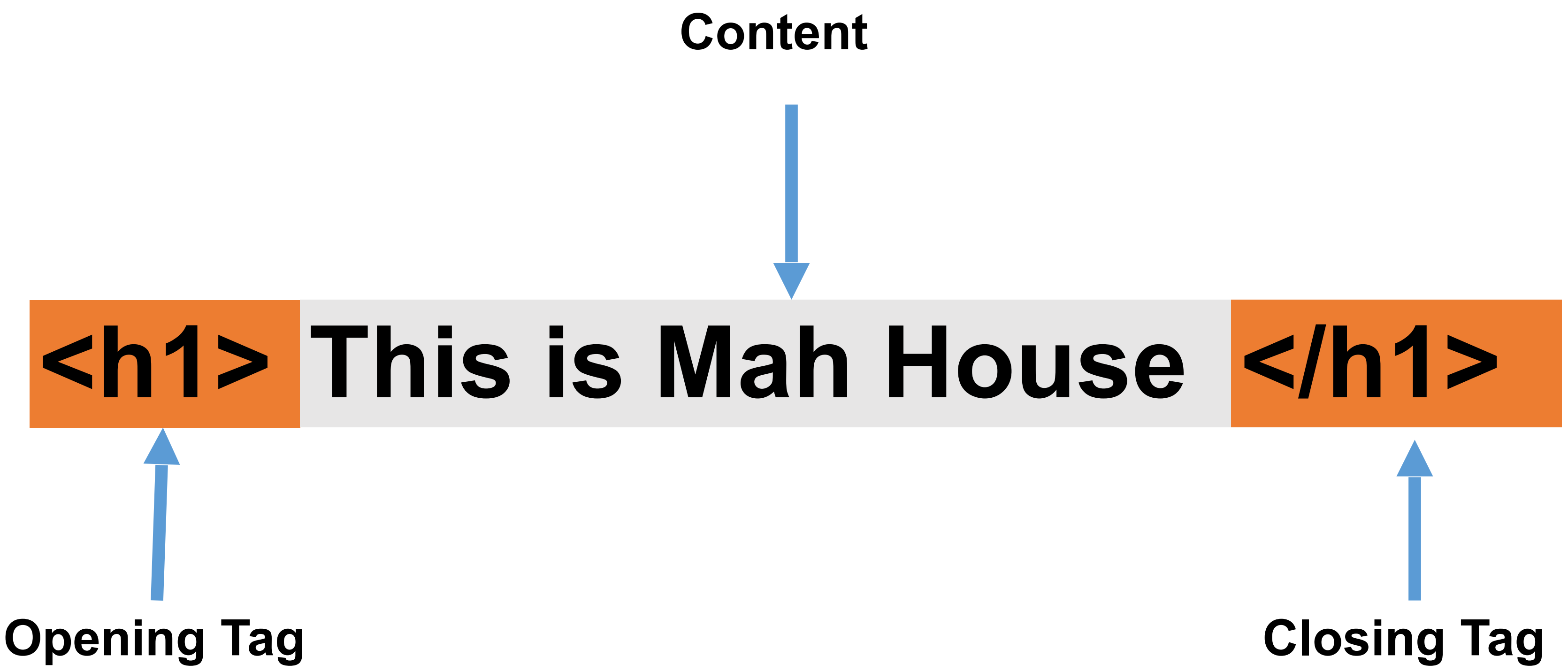
Gregg Pollack

Gregg is passionate about taking complex topics and teaching them effectively. He's helped build Envy Labs, Starter Studio, and Code School. He also furthers education through BarCamp in Orlando, the Orlando Ruby Users Group, and the Orlando Tech Events newsletter.

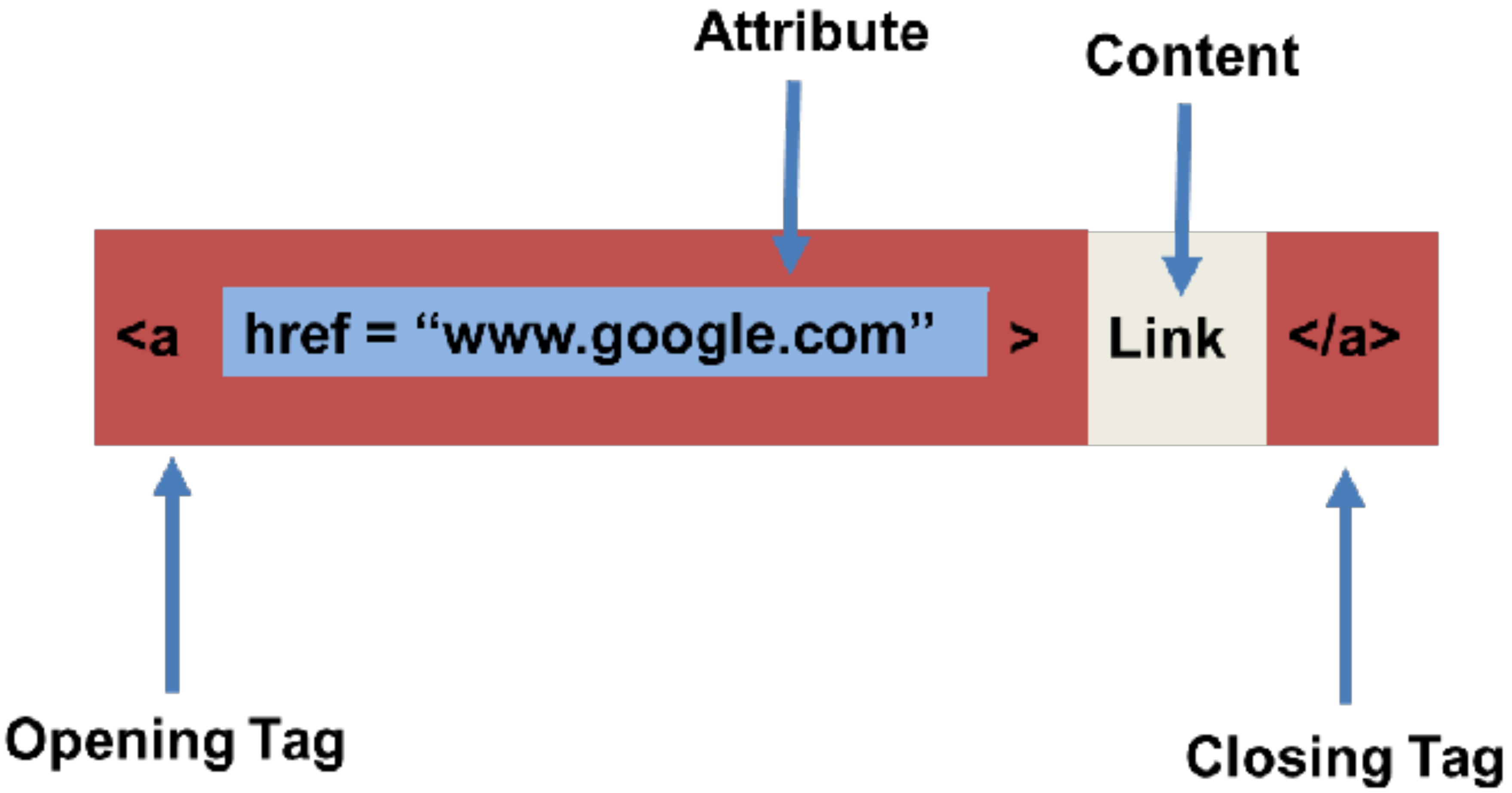
https://www.codeschool.com/courses/try-git

HTML Round 2

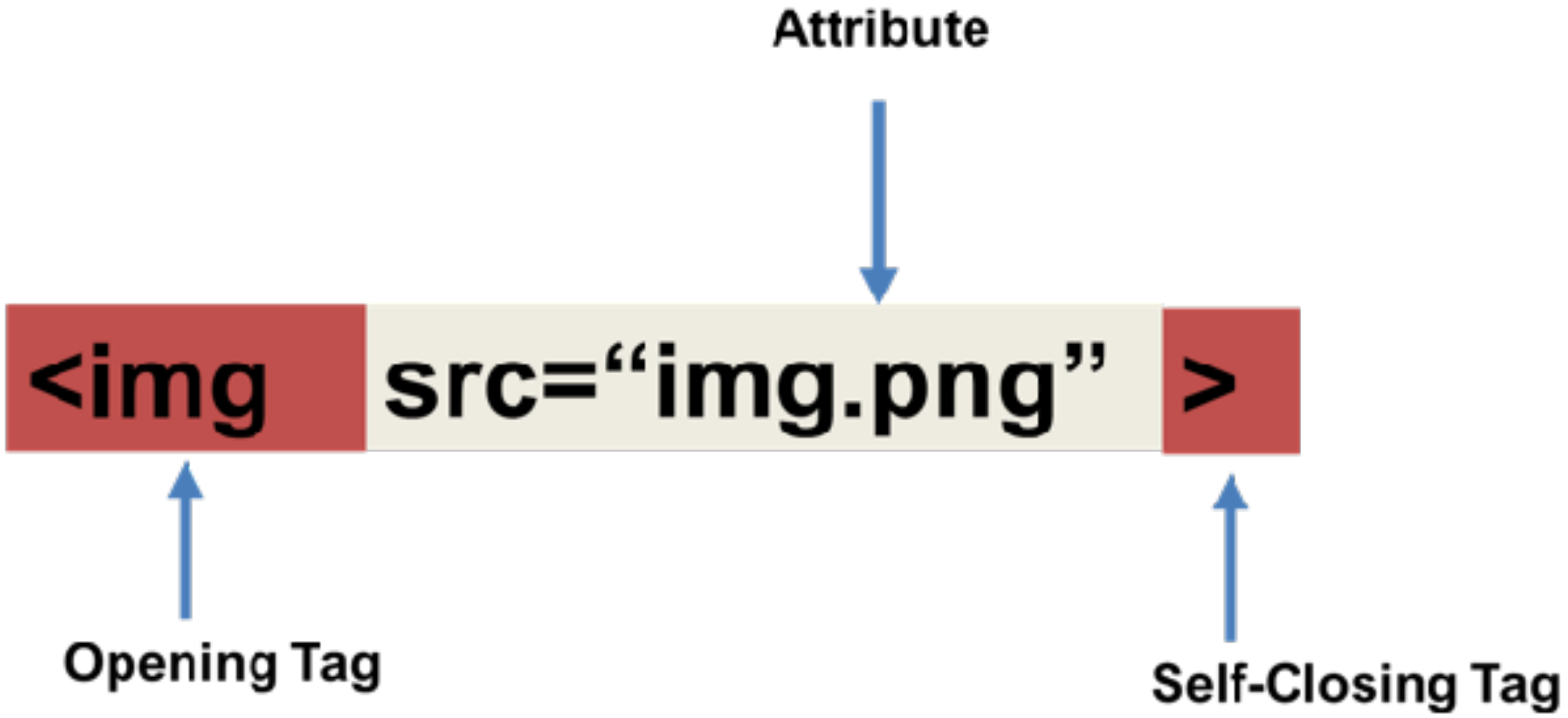
HTML Syntax (Basic)



HTML Syntax (with Attribute)



Tricky Tags (Self-Closing)



Important Common Tags

Headings:

- **<h1> </h1>** - Heading 1 (Largest Heading)
- **<h2> </h2>** - Heading 2 (Next Largest Heading)
- **<h3> </h3>** - Heading 3
- **…**

Containers:

- **<html> </html>** - Wraps the entire page
- **<head> </head>** - Wraps the header of the page
- **<body> </body>** - Wraps the main content
- **<div> </div>** - Logical Container ***
- **<p> </p>** - Wraps individual Paragraphs

Others:

- **<strong>** (bold), **<em>** (emphasis)
- **<img>** (images)**, <a href>** (links)**, <li>** (list items) **, <title>** (title), **<br>** (line break), **<table>** (tables), **<!-- -->** (comments)

Less Common Tags

- **All HTML Tags are listed here:** http://www.w3schools.com/tags/

- Don’t try to memorize them! Simply refer back to documentation as needed.

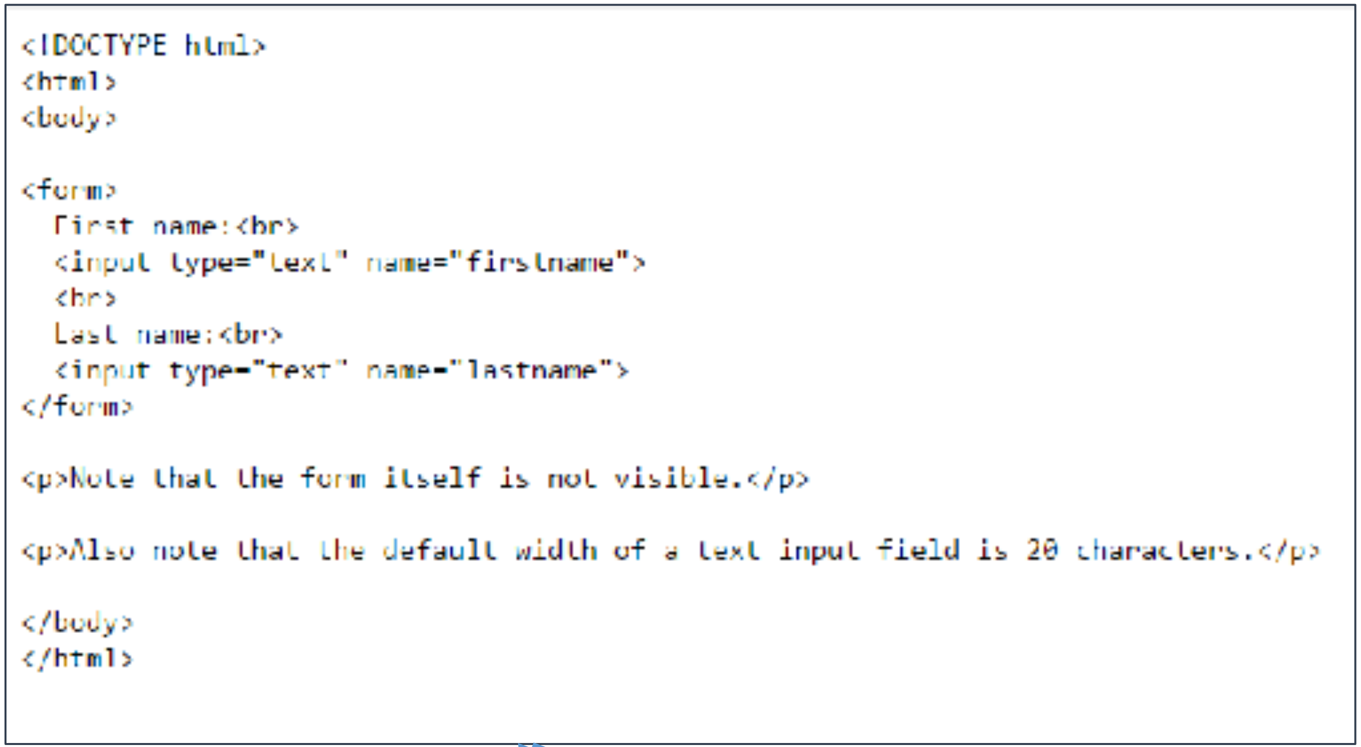
- Other tags:
  - <video> for Videos
  - <audio> for Audio files
  - <embed> for Embedded files
  - <code> for including computer code
  - <header> for headers
  - <nav> for navigation bars
  - <footer> for footers

HTML for Forms

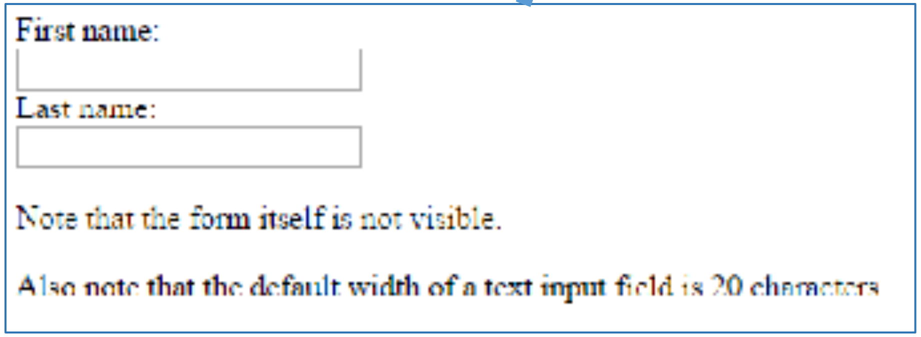
Common UI (User Interface) Form Elements:

- **<form>** - Creates a form section in HTML
- **<input>** - Input boxes
- **<label>** - Labels for boxes
- **<button>** - Button
- **<textarea>** - Large textbox

HTML for Forms



```
<!DOCTYPE html>
<html>
<body>

<form>
  First name:<br>
  <input type="text" name="firstname">
  <br>
  Last name:<br>
  <input type="text" name="lastname">
</form>

<p>Note that the form itself is not visible.</p>

<p>Also note that the default width of a text input field is 20 characters.</p>

</body>
</html>
```



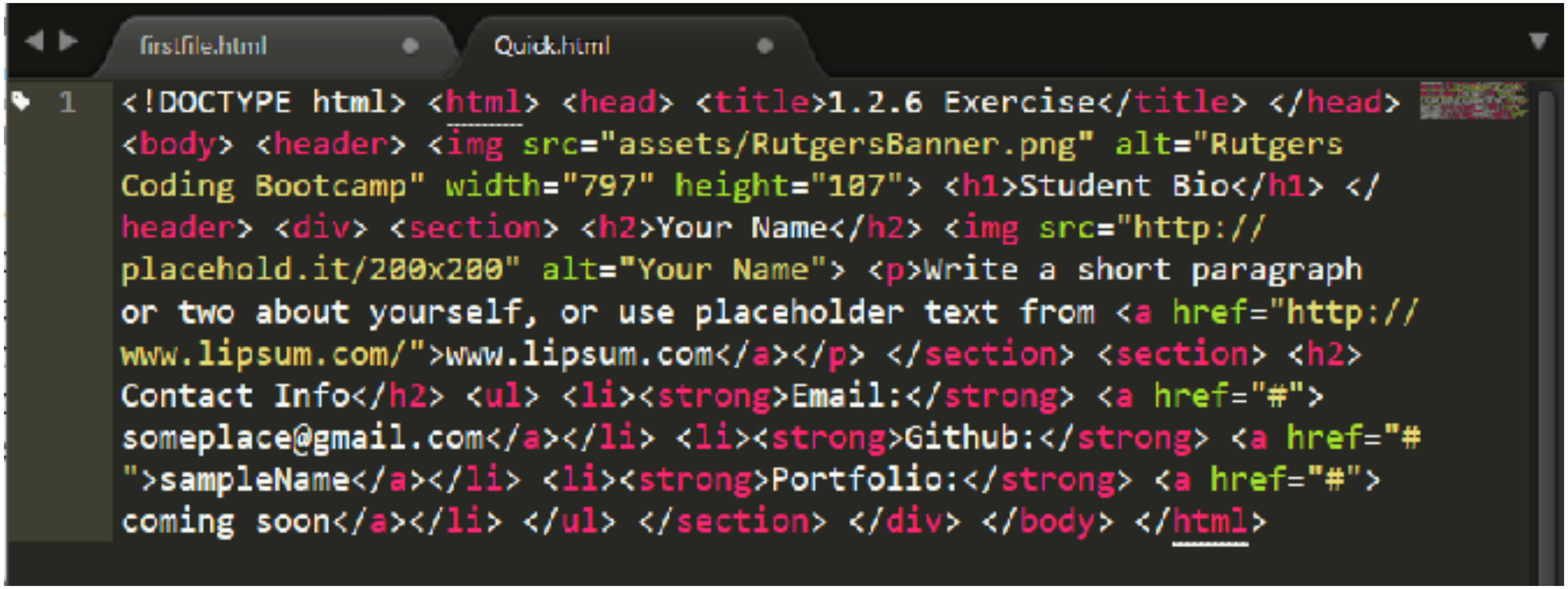
First name:

Last name:

Note that the form itself is not visible.

Also note that the default width of a text input field is 20 characters

On Ugly HTML



```
<!DOCTYPE html> <html> <head> <title>1.2.6 Exercise</title> </head>
<body> <header> <img src="assets/RutgersBanner.png" alt="Rutgers
Coding Bootcamp" width="797" height="107"> <h1>Student Bio</h1> </
header> <div> <section> <h2>Your Name</h2> <img src="http://
placehold.it/200x200" alt="Your Name"> <p>Write a short paragraph
or two about yourself, or use placeholder text from <a href="http://
www.lipsum.com/">www.lipsum.com</a></p> </section> <section> <h2>
Contact Info</h2> <ul> <li><strong>Email:</strong> <a href="#">
someplace@gmail.com</a></li> <li><strong>Github:</strong> <a href="#
">sampleName</a></li> <li><strong>Portfolio:</strong> <a href="#">
coming soon</a></li> </ul> </section> </div> </body> </html>
```

- Don't do this… Use proper indentation and sectioning.
- Readable code is easier to maintain.
- Invest time to get better about this now. It will pay dividends!

> YOUR TURN!

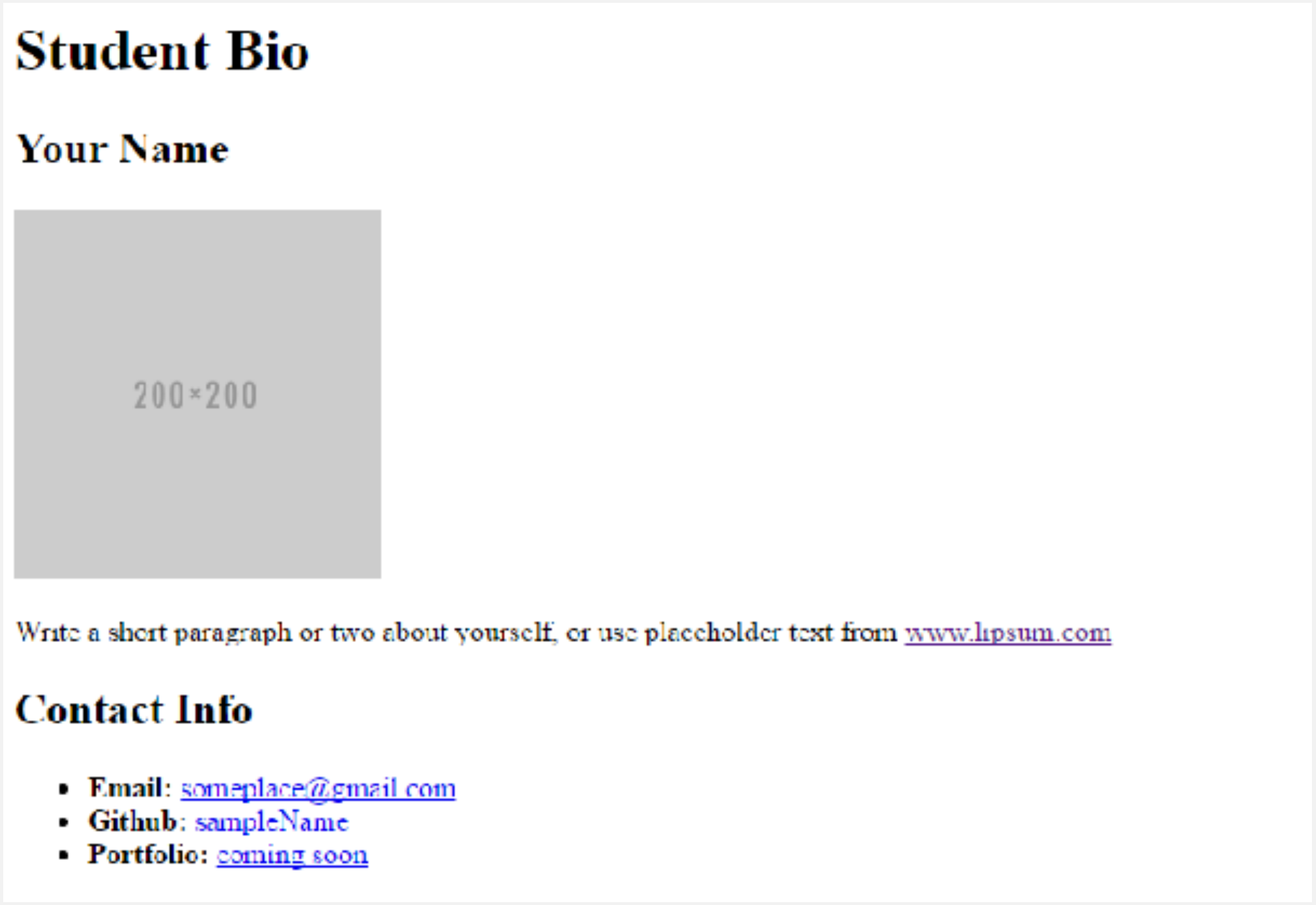
Activity: 1-HTML_Git | **Suggested Time:** 20 min

Assignment

In this activity, you'll create a student bio using HTML. You will then add, commit, and push your completed HTML to GitHub for the world to see.

Additional instructions, sent via Slack.

> YOUR TURN!



Student Bio

Your Name

Write a short paragraph or two about yourself, or use placeholder text from www.lipsum.com

Contact Info

- **Email:** someplace@gmail.com
- **Github:** sampleName
- **Portfolio:** coming soon

CSS Stylin’

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

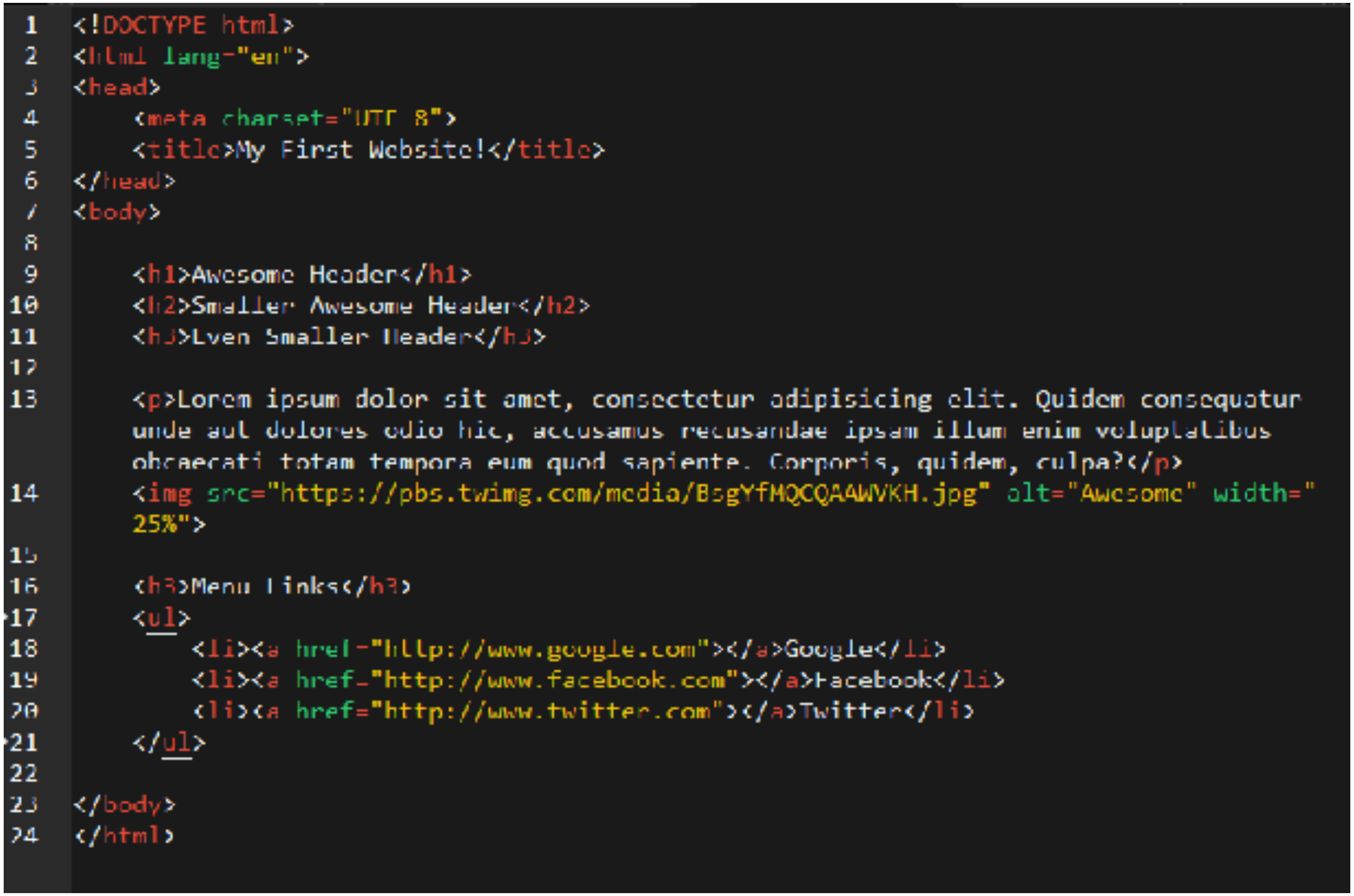
HTML Alone

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

HTML / CSS

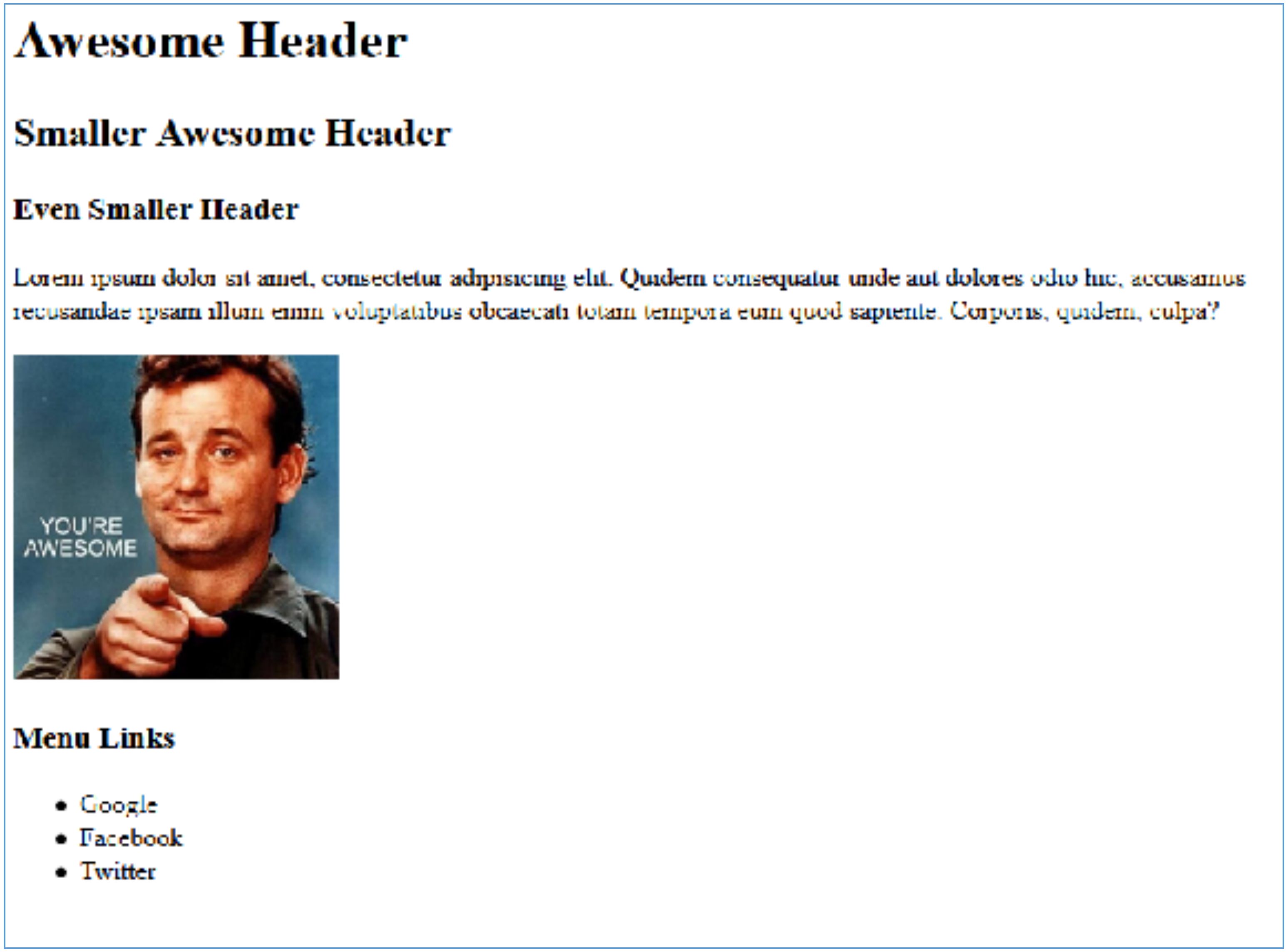
- Like writing papers in Microsoft Word.
- Can format text, page settings, alignment, etc. based on "highlighting" and menu options.

Basic HTML Page



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>My First Website!</title>
</head>
<body>

    <h1>Awesome Header</h1>
    <h2>Smaller Awesome Header</h2>
    <h3>Even Smaller Header</h3>

    <p>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?</p>
    <img src="https://pbs.twimg.com/media/BsgYfMQCQAAWVKH.jpg" alt="Awesome" width="
    25%">

    <h3>Menu Links</h3>
    <ul>
        <li><a href="http://www.google.com"></a>Google</li>
        <li><a href="http://www.facebook.com"></a>Facebook</li>
        <li><a href="http://www.twitter.com"></a>Twitter</li>
    </ul>

</body>
</html>
```

Basic HTML Page - Result



Awesome Header

Smaller Awesome Header

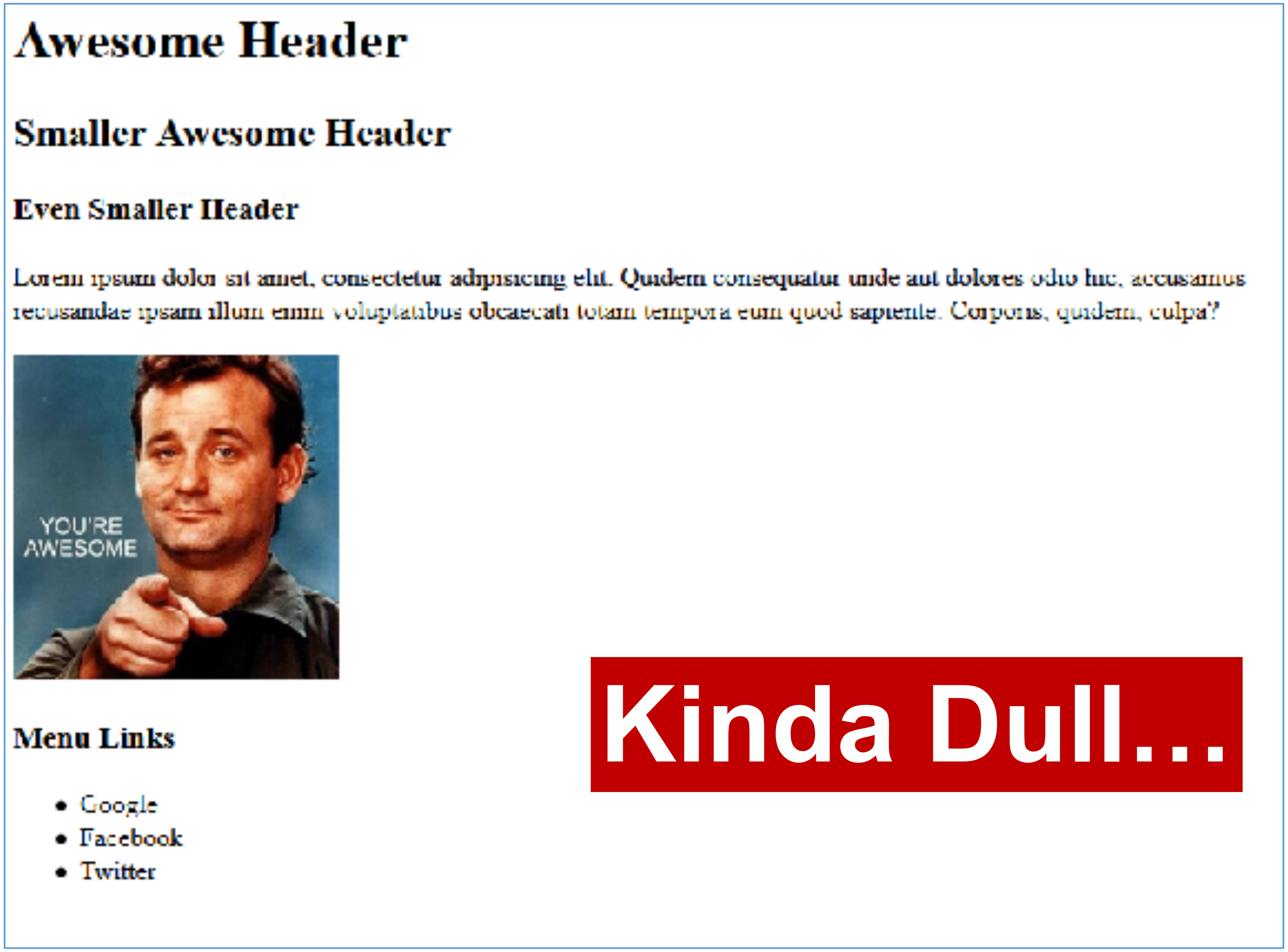
Even Smaller Header

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

Basic HTML Page - Result



Awesome Header

Smaller Awesome Header

Even Smaller Header

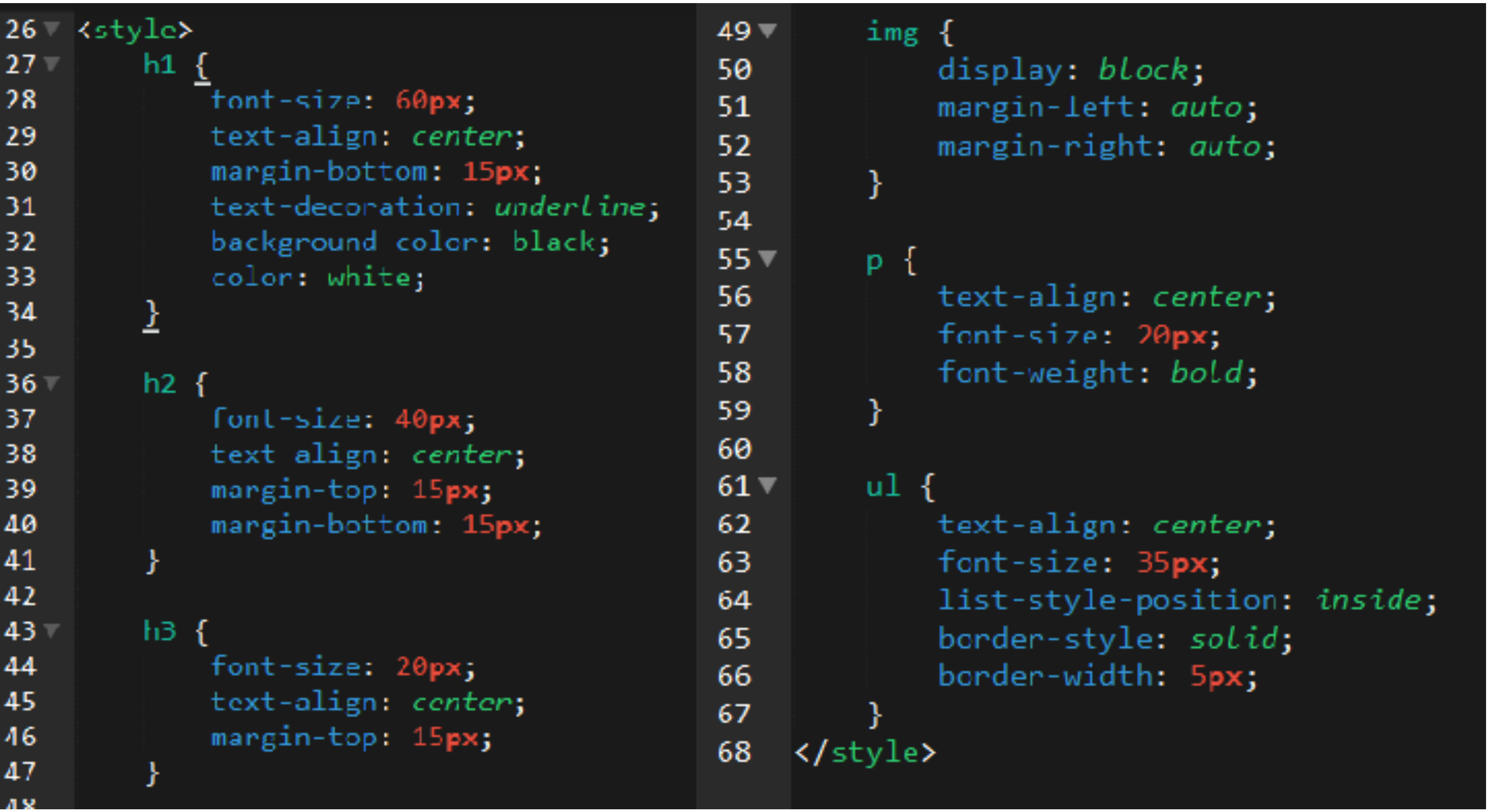
Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quidem consequatur unde aut dolores odio hic, accusamus recusandae ipsam illum enim voluptatibus obcaecati totam tempora eum quod sapiente. Corporis, quidem, culpa?

Menu Links

- Google
- Facebook
- Twitter

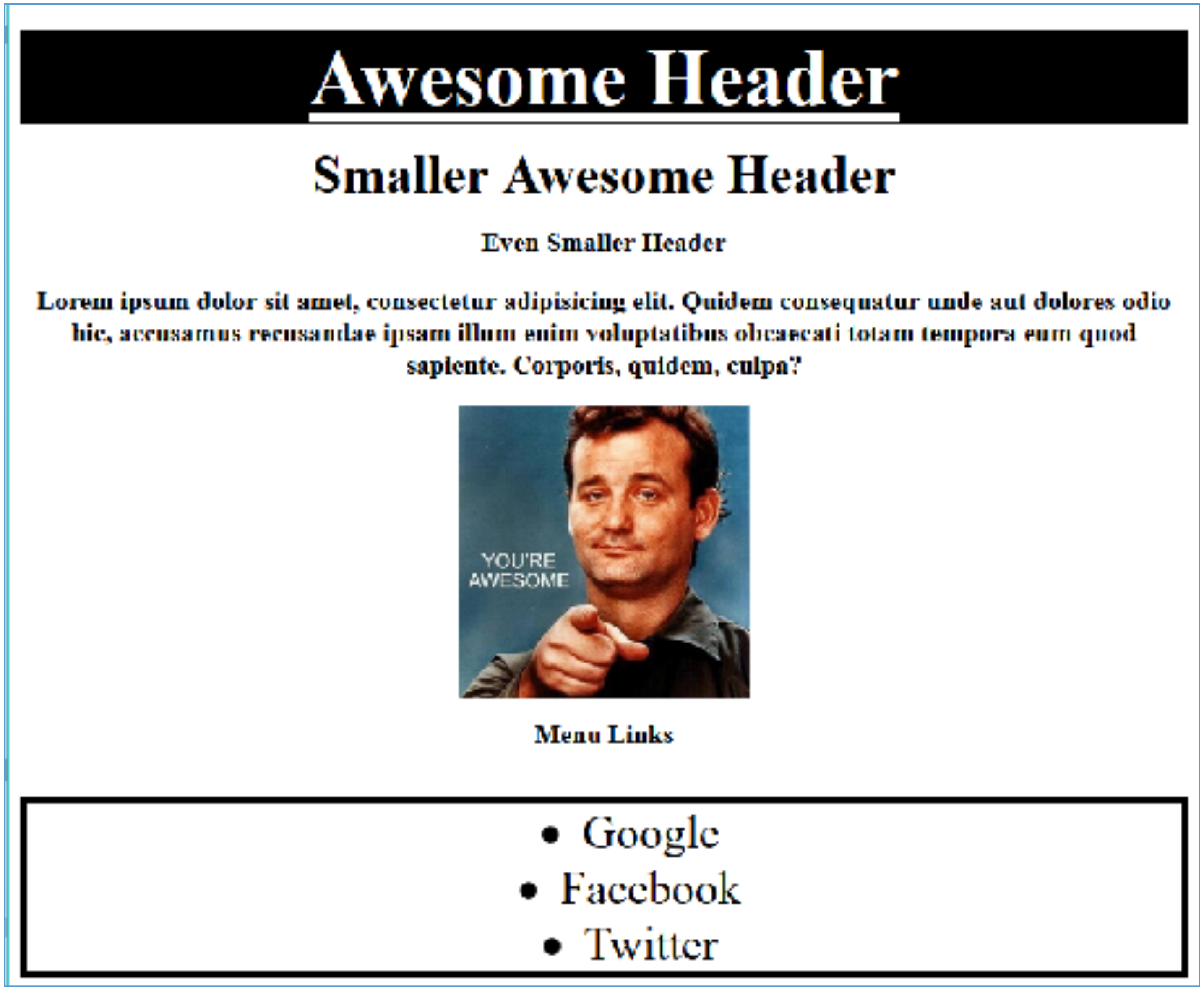
Kinda Dull…

Enter CSS



```
<style>
    h1 {
        font-size: 60px;
        text-align: center;
        margin-bottom: 15px;
        text-decoration: underline;
        background color: black;
        color: white;
    }

    h2 {
        font-size: 40px;
        text align: center;
        margin-top: 15px;
        margin-bottom: 15px;
    }

    h3 {
        font-size: 20px;
        text-align: center;
        margin-top: 15px;
    }

    img {
        display: block;
        margin-left: auto;
        margin-right: auto;
    }

    p {
        text-align: center;
        font-size: 20px;
        font-weight: bold;
    }

    ul {
        text-align: center;
        font-size: 35px;
        list-style-position: inside;
        border-style: solid;
        border-width: 5px;
    }
</style>
```

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?

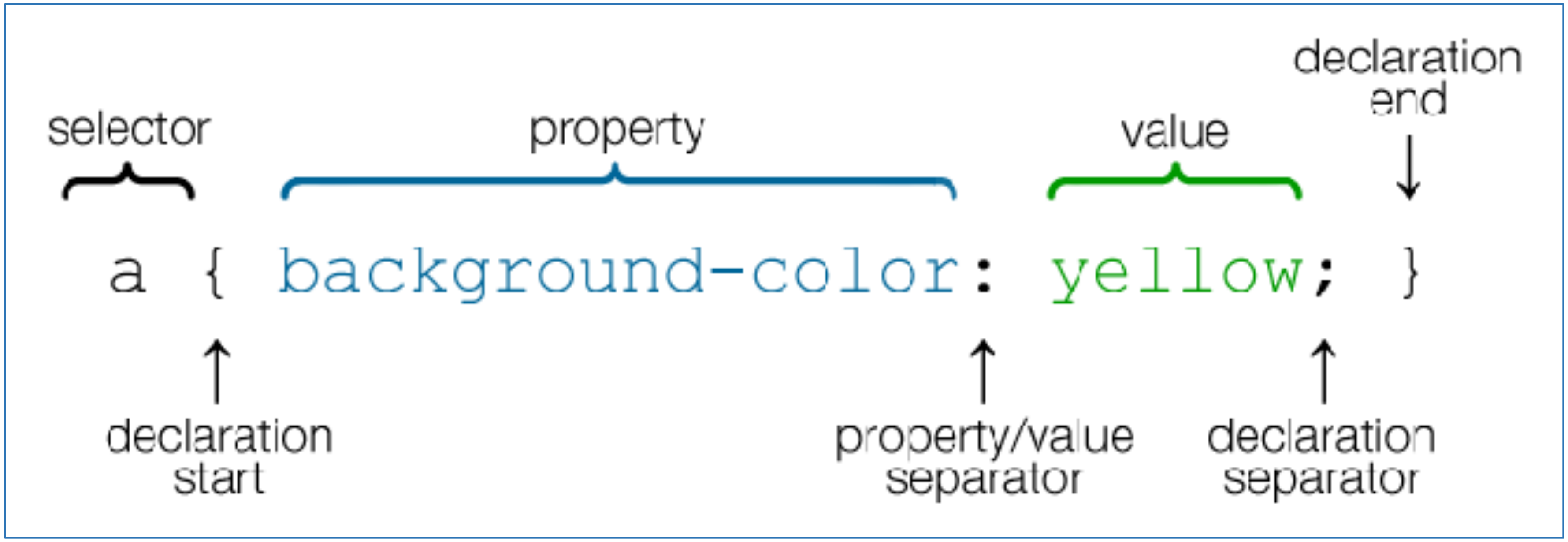
Menu Links

- Google
- Facebook
- Twitter

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.


```
a { background-color: yellow; }
```

CSS Example

- In the below example the “Header” would be turned blue and MUCH larger because of the CSS.
- We can incorporate an element’s class or ID to apply a CSS style to a particular part of the document.
  - Just remember to include the necessary symbol before the CSS: “.” for class, “#” for ID.

Example (HTML):

```
<p class=“bigBlue”>Header</p>
<p id=“smallRed”>This text is tiny</p>
```

Example (CSS):

```
.bigBlue {
    font-size: 100px;
    color: blue;
}

#smallRed {
    font-size: 8px;
    color: red;
}
```

Key CSS Attributes

Font / Color:

- **color**: Sets color of text.
- **font-size**: Sets size of the font.
- **font-style**: Sets italics.
- **font-weight**: Sets bold.

Alignment / Spacing:

- **padding (top/right/bottom/left):** Adds space between element and its own border.

- **margin (top/right/bottom/left):** Adds space between element and surrounding elements.

- **float:** Forces elements to the sides.

Background:

- **background-color:** sets background color.
- **background-image:** sets background image.

Powerful Duo

Believe it or not, HTML / CSS is all you need to develop a vivid, full-blown website.

INSTRUCTOR DEMO

Instructor: Demo

(quickexample_internalcss.html | 2-BasicCSS)

> YOUR TURN!

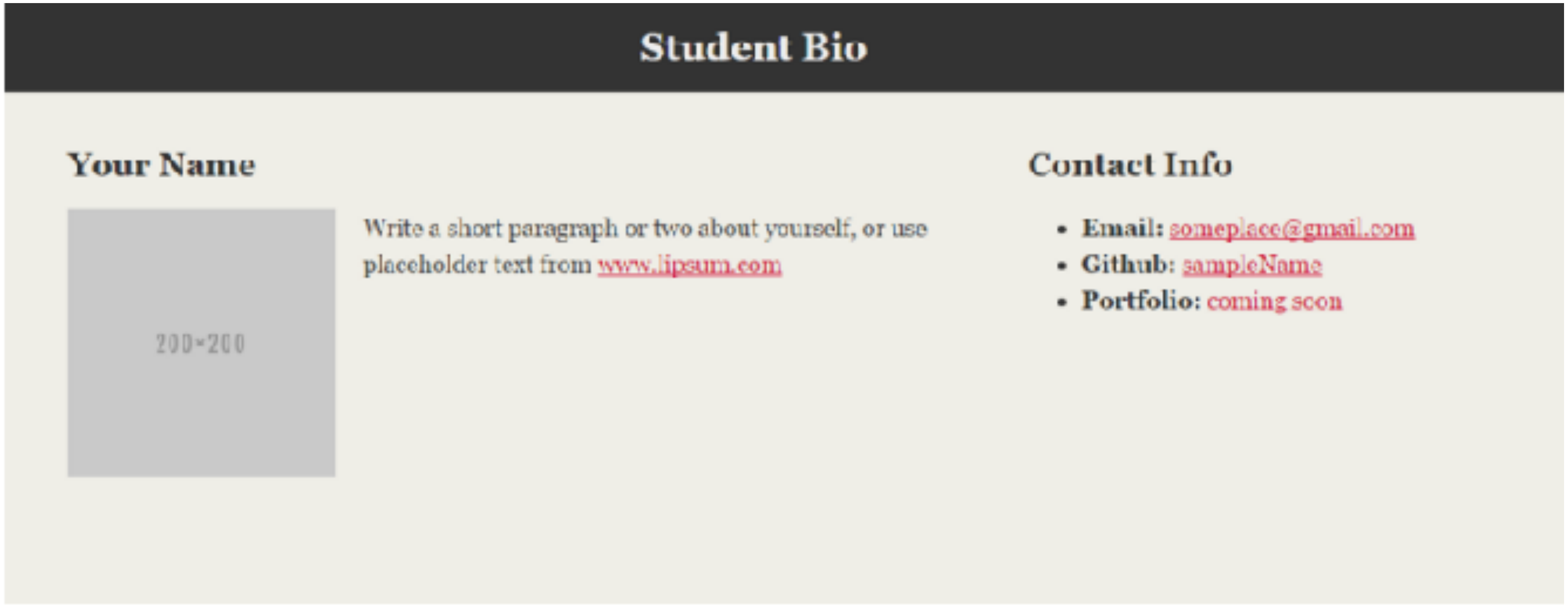
Activity*:* 3-HTML_CSS_Layout **| Suggested Time:** 20 min

Assignment

In this activity, you’ll upgrade your previous HTML bio-page using CSS style rules. Once you’re done, commit and push up your changes to GitHub.

We’ll send you additional instructions via Slack.

> YOUR TURN!



Student Bio

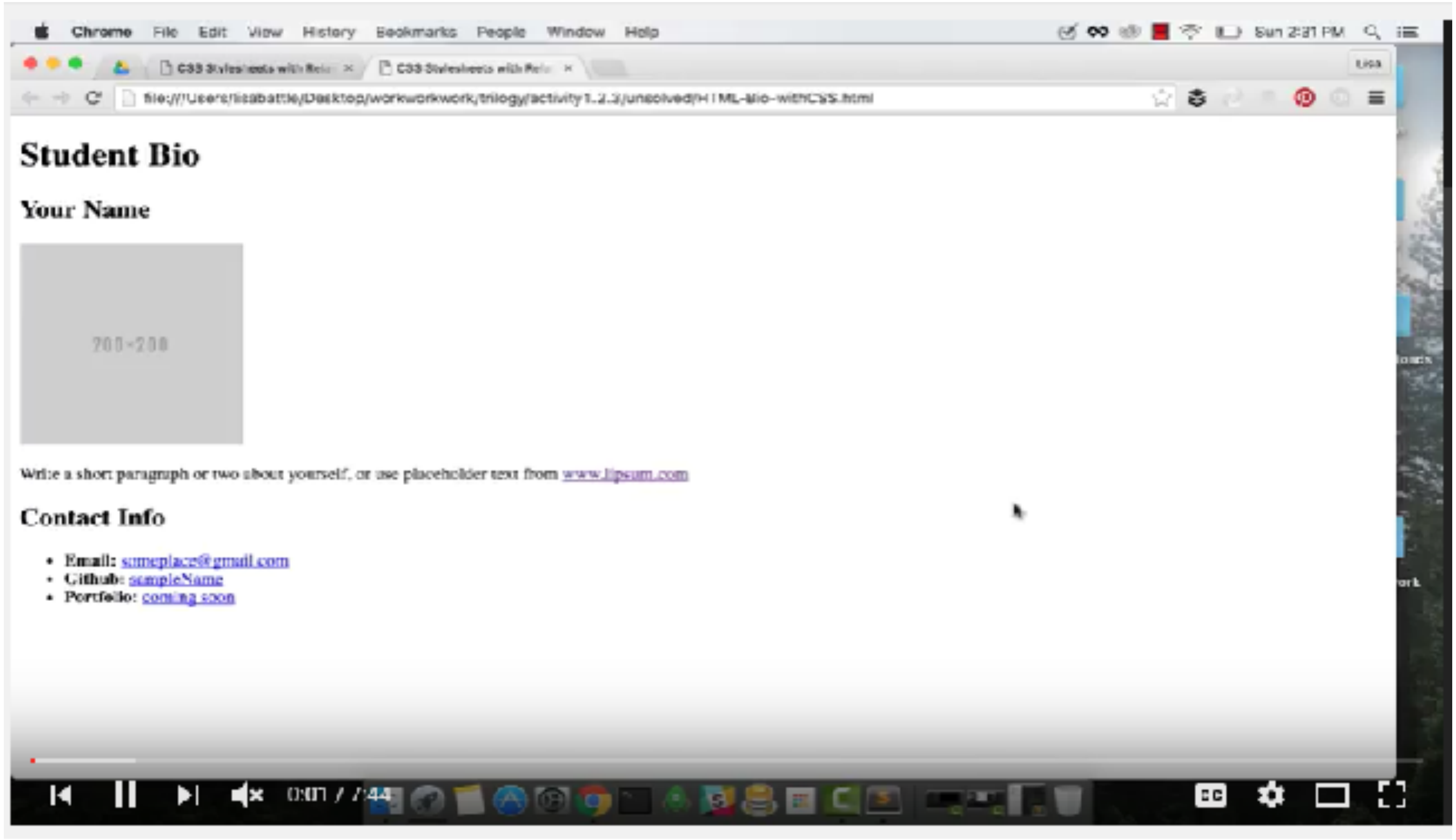
Your Name

Write a short paragraph or two about yourself, or use placeholder text from www.lipsum.com

Contact Info

- **Email:** someplace@gmail.com
- **Github:** sampleName
- **Portfolio:** coming soon

Video Walkthrough!!


Chrome File Edit View History Bookmarks People Window Help
Sun 2:31 PM
file:///Users/lisabattle/Desktop/workworkwork/trilogy/activity 1.2.3/unsolved/HTML-Bio-withCSS.html
0:07 / 7:44

Student Bio

Your Name

200×200

Write a short paragraph or two about yourself, or use placeholder text from www.lipsum.com

Contact Info

- Email: someplace@gmail.com
- Github: sampleName
- Portfolio: coming soon

https://www.youtube.com/watch?v=kMBinXTCrXI&list=PLgJ8UgkiorCnMLsUevoQRxH8t9bt7ne14&index=2

Still a Bit Confused?

Remember! We've got video guides for key activities like that last one.

If you feel like you are EVER falling behind, use those online walkthroughs to help catch back up. They are made to be easy to understand.

Still having trouble? Shoot your instructor or one of your TAs a message!
We are here to help you out in whatever way we can!

Recap + Questions