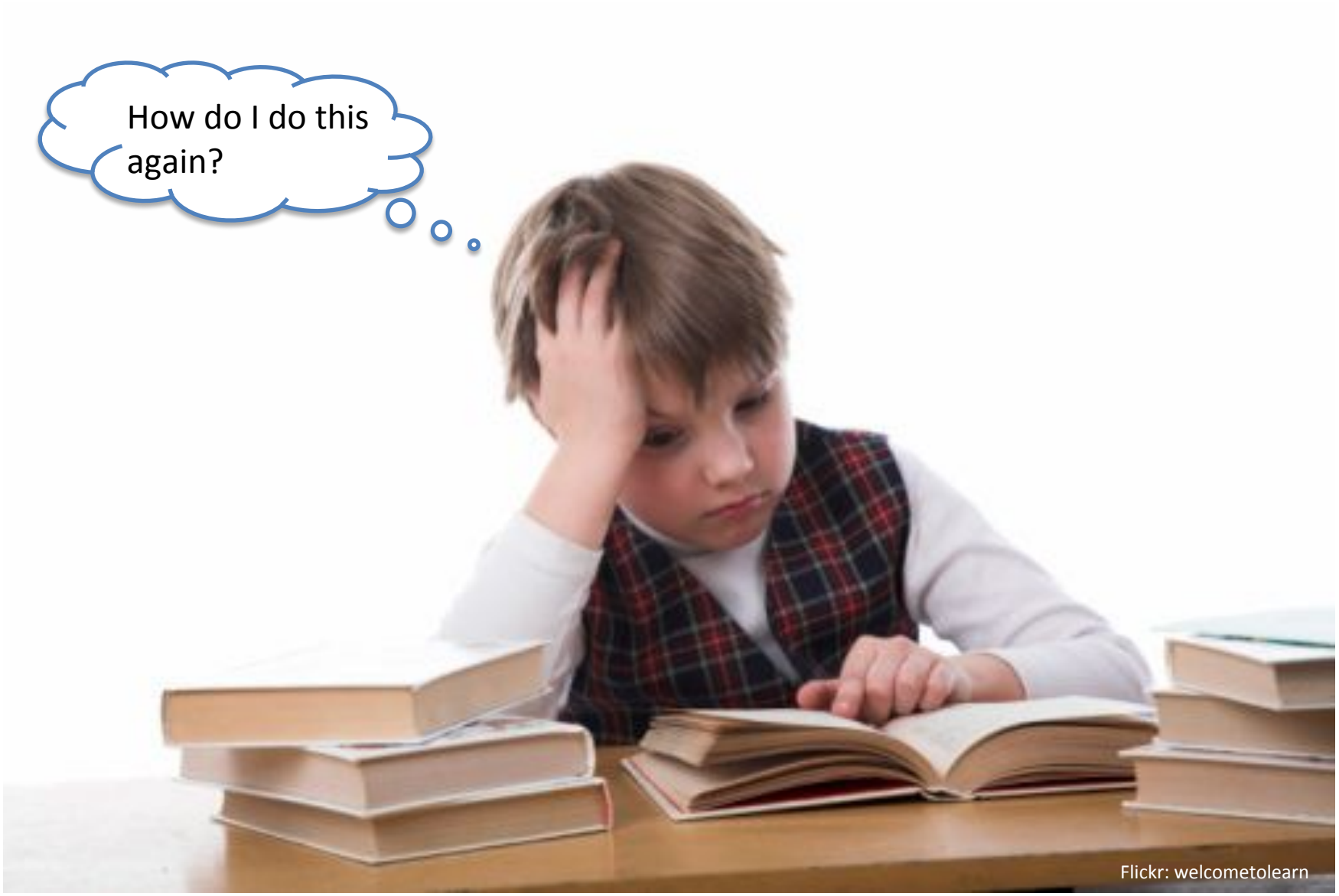
Git'n Pro with HTML/CSS

The Coding Bootcamp

It’s Okay!



Admin Items

Where to Get Help

- **Practice, Practice, Practice:** Work Individually or in Groups
- **Review In Class Material (Exercises and Slides):** https://upenn.bootcampcontent.com/upenn-bootcamp/UPENN201901FSF5
- **Re-Watch Class Videos:** https://www.bootcampspot.com/login
- **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 class content and homework repository.**

  https://upenn.bootcampcontent.com/upenn-bootcamp/UPENN201901FSF5

- **Homework Assignment #1 is due next week, 2/9**

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:

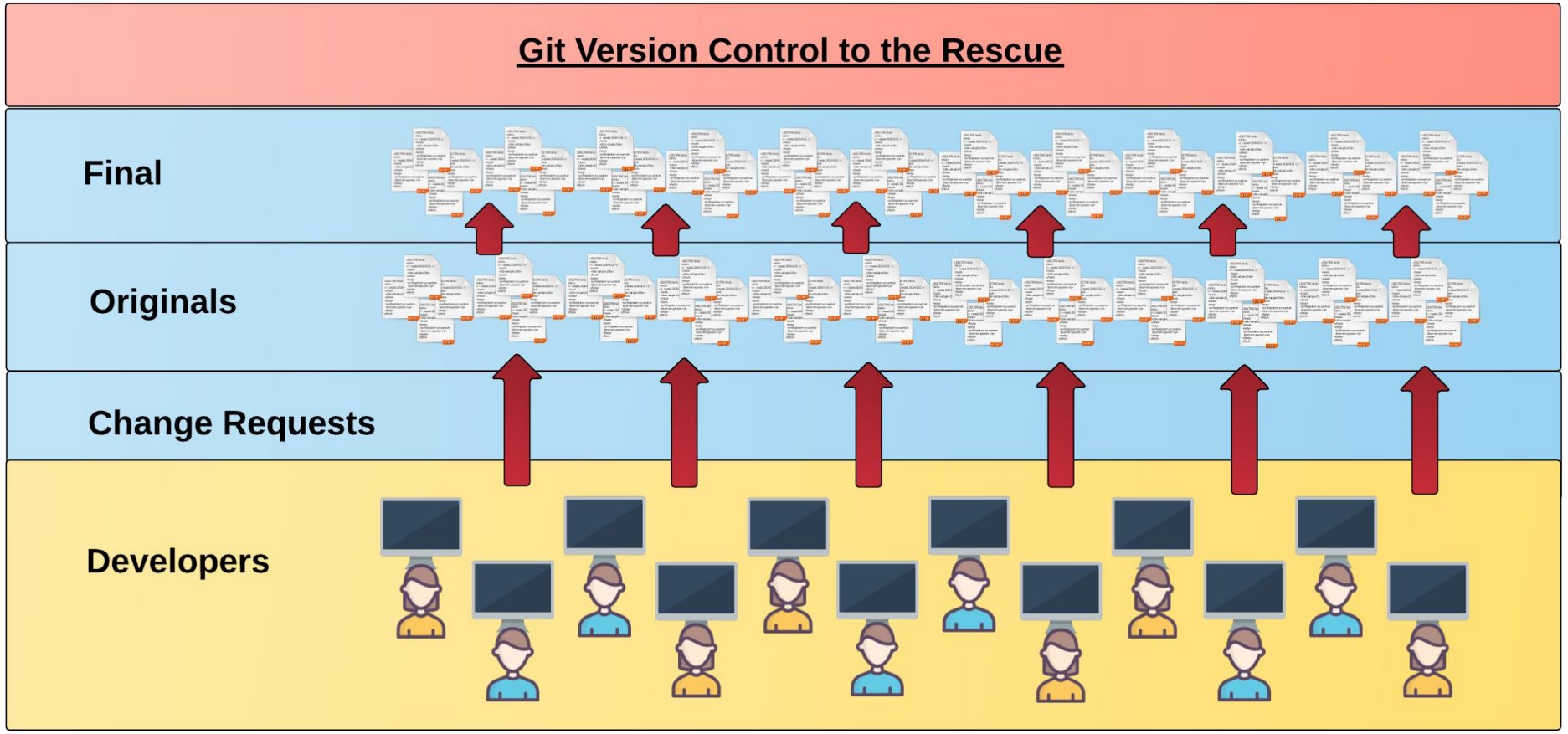
- Continue getting comfortable with HTML.
- Be able to completely write a basic HTML document (like in last class).
- Understand what CSS is, what it's for, and how it works with HTML.
- *Be able to use Git and GitHub to upload code.*

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 *highly* collaborative.
- Teams are often extremely large and separated across the country — or planet.
- Apps sometimes comprise hundreds or even thousands of files.

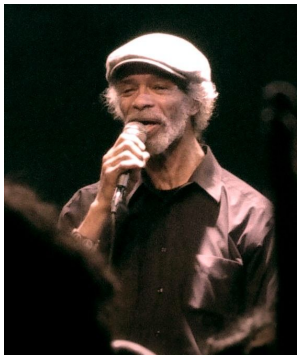
The Team's Task

Task: Make a list of creative works you've written in the past

Programming Team:

Maya Angelou

Anne Sexton

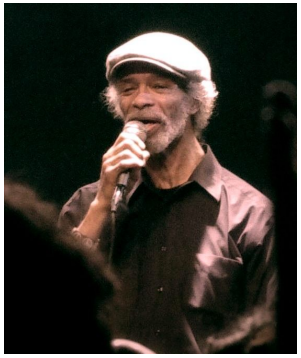


Gil Scott Heron

Maya & Gil make their edits

Programming Away…

Maya's Version

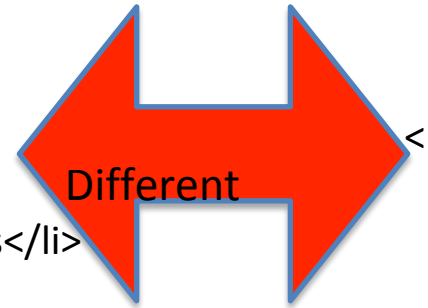


Programming Away…

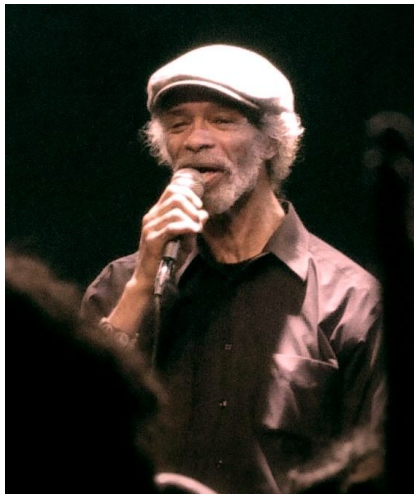
Gil's Version

Different Solutions

```
<ul>
   <li>On the Pulse of Morning</li>
   <li>I Know Why the Caged Bird Sings</li>
   <li>And Still I Rise</li>
</ul>
```




```
<ul>
   <li>Free Will</li>
   <li>Pieces of a Man</li>
   <li>The Revolution will not be Televised</li>
</ul>
```

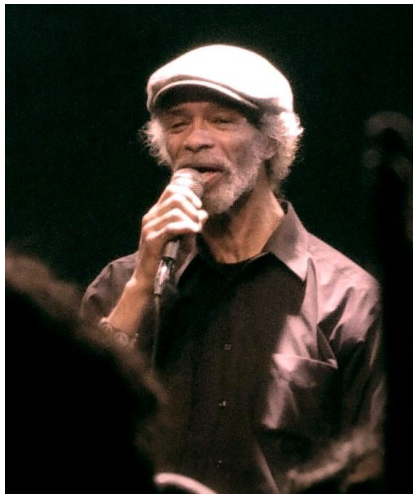


Resolution

```
<ul>
   <li>On the Pulse of Morning</li>
   <li>I Know Why the Caged Bird Sings</li>
   <li>And Still I Rise</li>
</ul>
```

“Let’s settle on this…”

```
<ul>
   <li>Poems</li>
   <li>Albums</li>
   <li>Songs</li>
</ul>
```



```
<ul>
   <li>Free Will</li>
   <li>Pieces of a Man</li>
   <li>The Revolution will not be Televised</li>
</ul>
```

Anne writes her own stuff…

```
<ul>
    <li>The Double Image</li>
    <li>Heart's Needle</li>
    <li>Baby Picture</li>
</ul>
```

Anne overwrites work of her teammates…

Delete. Delete.
Delete. Delete.
Delete. Delete

~~<ul>~~
~~
Poems
~~
~~
Albums
~~
~~
Songs
~~
~~
~~

The Double Image
45 Mercy Street
The Road Back

The Group Project

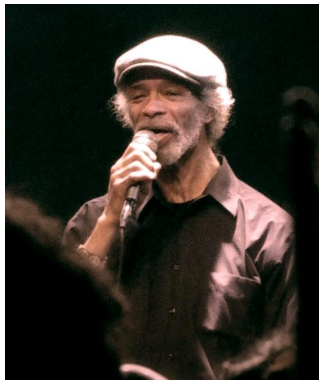
Lesson:

You should use Version Control.

….and watch your teammates' work

Today we fret and pull
on wheels, ignore our regular loss
of time…

…or maybe we should just use git



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

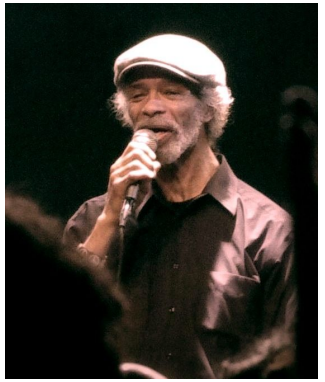
Master Branch

'Branch' = personal copy

Personal branch

Maya's branch

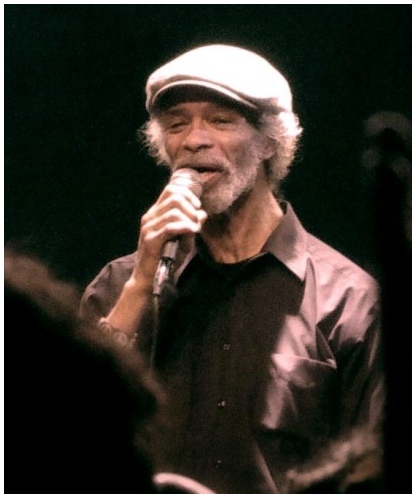
Anne's branch



Gil's branch

The team goes to work

```
<ul>
   <li>On the Pulse of Morning</li>
   <li>I Know Why the Caged Bird Sings</li>
   <li>And Still I Rise</li>
</ul>
```



```
   <ul>
      <li>Free Will</li>
      <li>Pieces of a Man</li>
      <li>The Revolution will not be Televised</li>
   </ul>
```

Maya pushes first

Master Copy

1

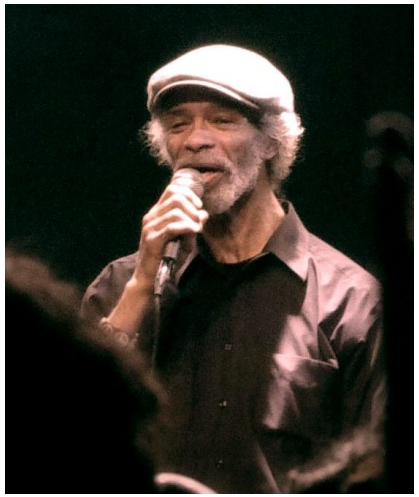
Maya's Branch

Maya **pushes (uploads)** her code changes into the main branch.

No code conflicts.

Gil's edits are ready

Rule: pull first, then push your changes



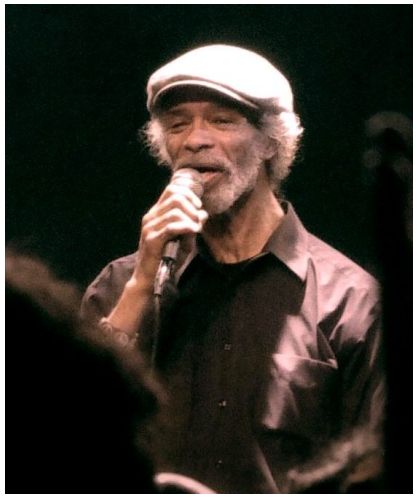
Ok

Gil pulls latest changes

Master Copy

1

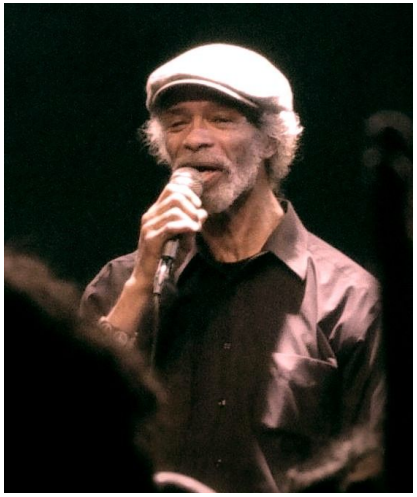
Gil's Branch



Gil conflicts with master branch

Master Branch

1



On the Pulse of Morning
I Know Why the Caged Bird Sings
And Still I Rise
Free Will
Pieces of a Man
The Revolution will not be Televised

Git sees a conflict.

Gil resolves

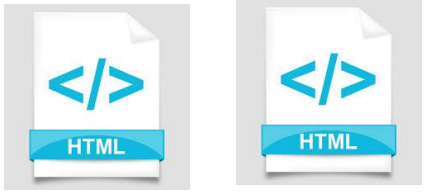
```
<li>On the Pulse of Morning</li>
<li>I Know Why the Caged Bird Sings</li>
<li>And Still I Rise</li>
<li>On the Pulse of Morning</li>
<li>I Know Why the Caged Bird Sings</li>
<li>And Still I Rise</li>
```

```
<ul>
   <li>Poems</li>
   <li>Albums</li>
   <li>Songs</li>
</ul>
```

Gil's Branch

Gil fixes and pushes

Master Branch



1 2

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

No code conflicts.

Gil's Branch

```
<ul>
   <li>Poems</li>
   <li>Albums</li>
   <li>Songs</li>
</ul>
```

Anne starts her work

Rule: pull first, then push your changes

look into my face
and you will know that crimes dropped upon me
as from a high building…

…by which I mean, I broke the rules.

Anne pushes

Master Branch

1 2 3

Anne dude **pushes (uploads)** her revision the main branch. No code conflicts.

Not what we want.

Anne's Branch

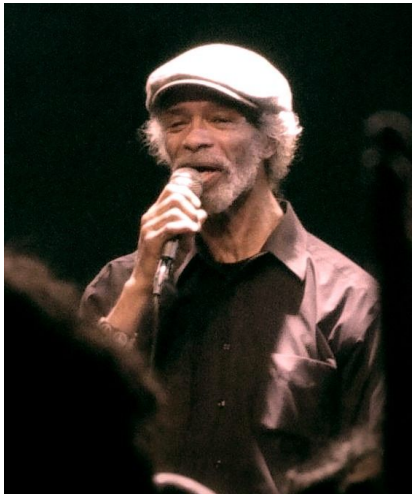
```
<ul>
   <li>The Double Image</li>
   <li>Heart's Needle</li>
   <li>Baby Picture</li>
</ul>
```

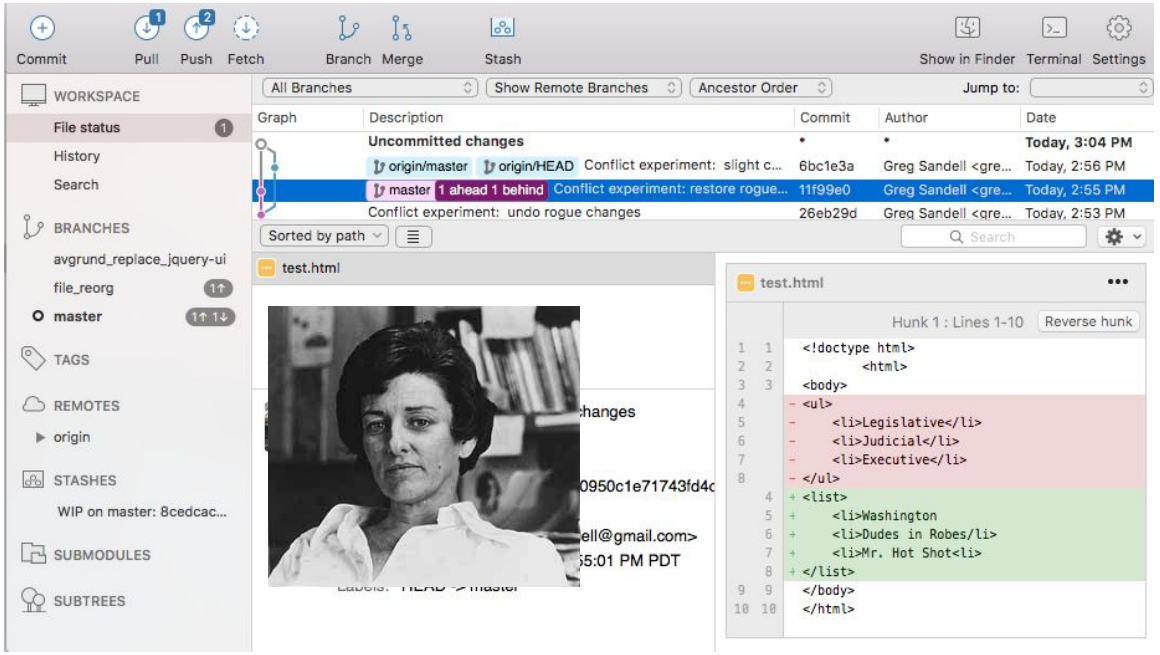
If Anne had made a pull first…

Conflict!

```
<ul>
   <li>The Double Image</li>
   <li>Heart's Needle</li>
   <li>Baby Picture</li>
</ul>
<ul>
   <li>Poems</li>
   <li>Albums</li>
   <li>Songs</li>
</ul>
```

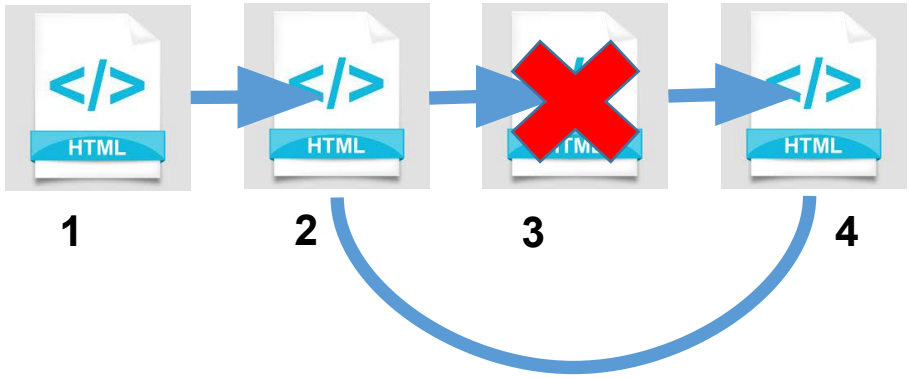
The overwritten work is discovered





Roll Back

Main Branch



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

Anne's Branch

Lesson:

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.**

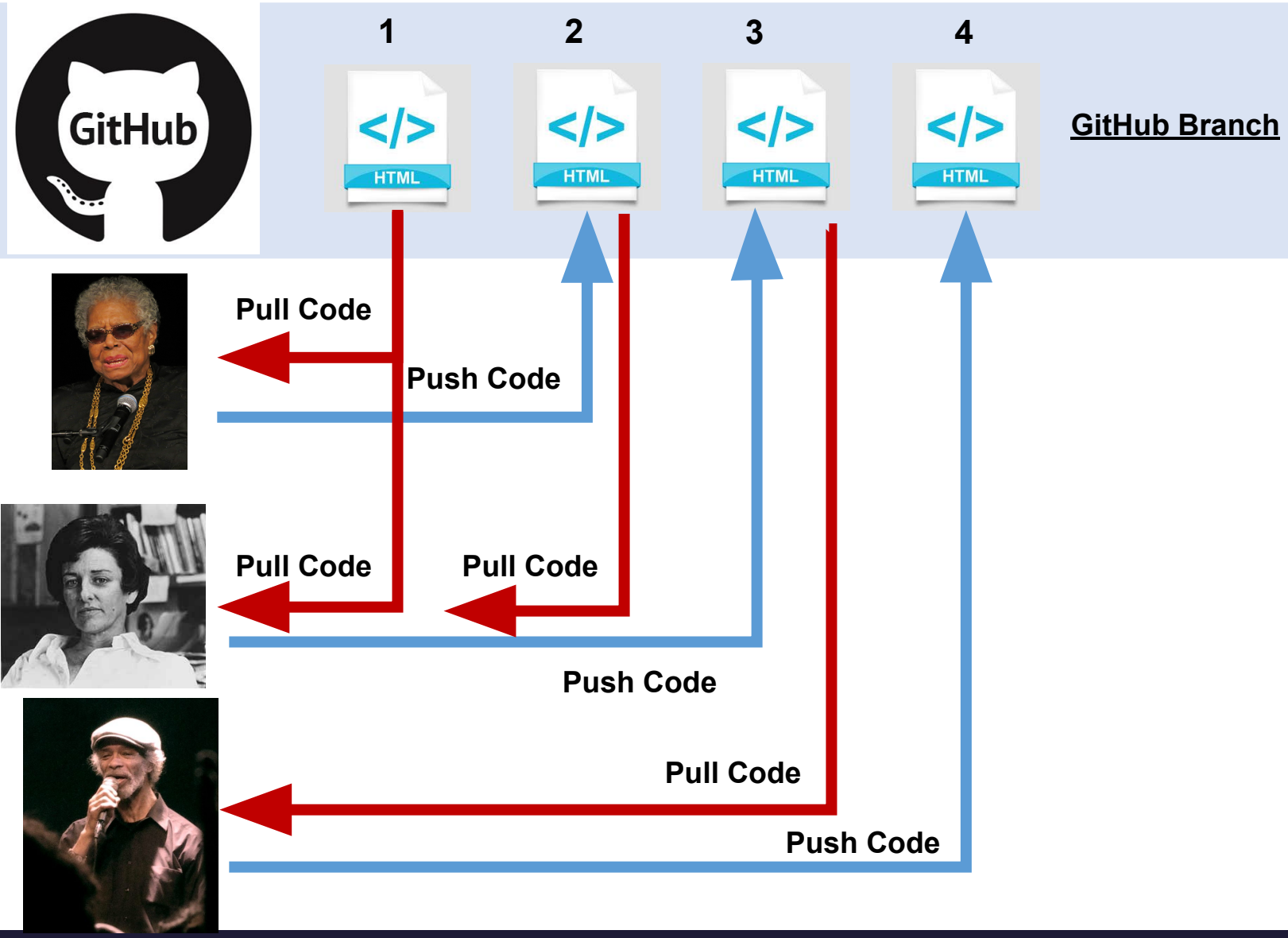
Then the other should explain:

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

So… What's this GitHub?

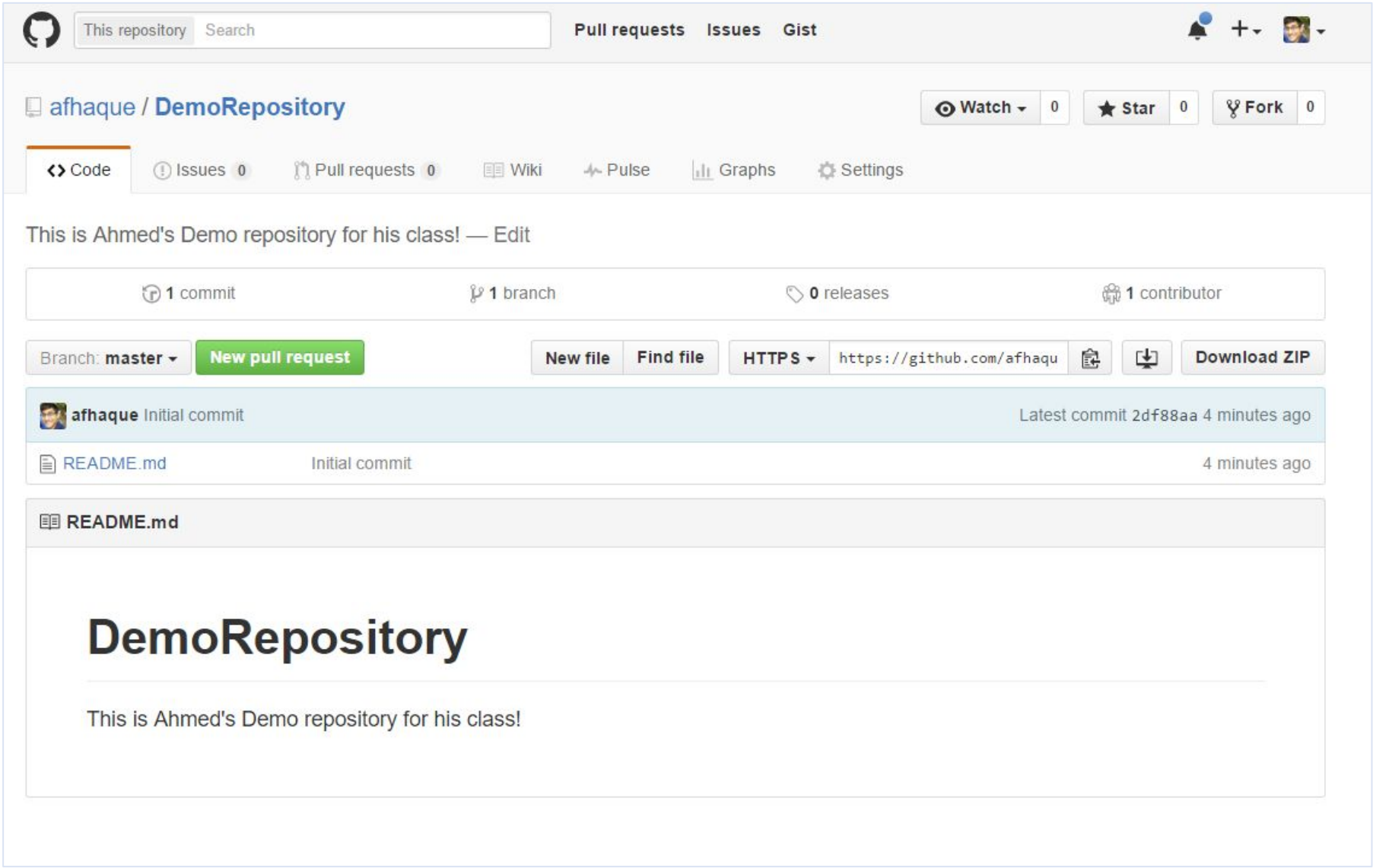
- GitHub is a Web-Based hosting service to store code online.
- It allows developers to **pull** (download) code or **push** (upload) code to the same **repository** (directory).
- It also allows developers to **view histories** of code changes and to **track issues**.

Pushing and Pulling to GitHub



Get Started with Git

Instructor Git Demo!



Basic Git Commands

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

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.

> 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:

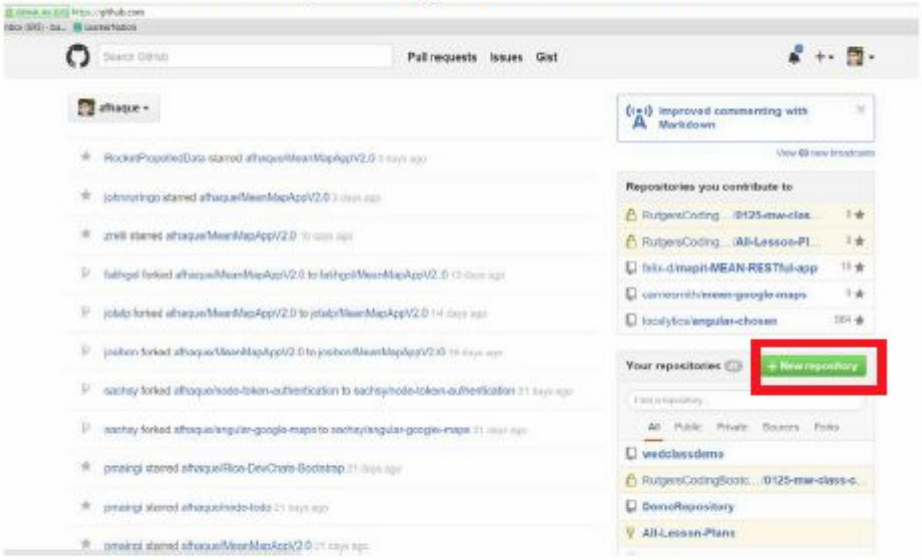
- Find a partner in class, and **fork** *their* repository to your own GitHub account. Clone this forked repository to your local directory.
- Add, Commit, and Push the code back to your forked copy.
- Finally, submit a **pull request** to send your changes to your partner’s repo.

Still a Bit Lost? Never Worry!

Steps to Uploading Your Code to GitHub

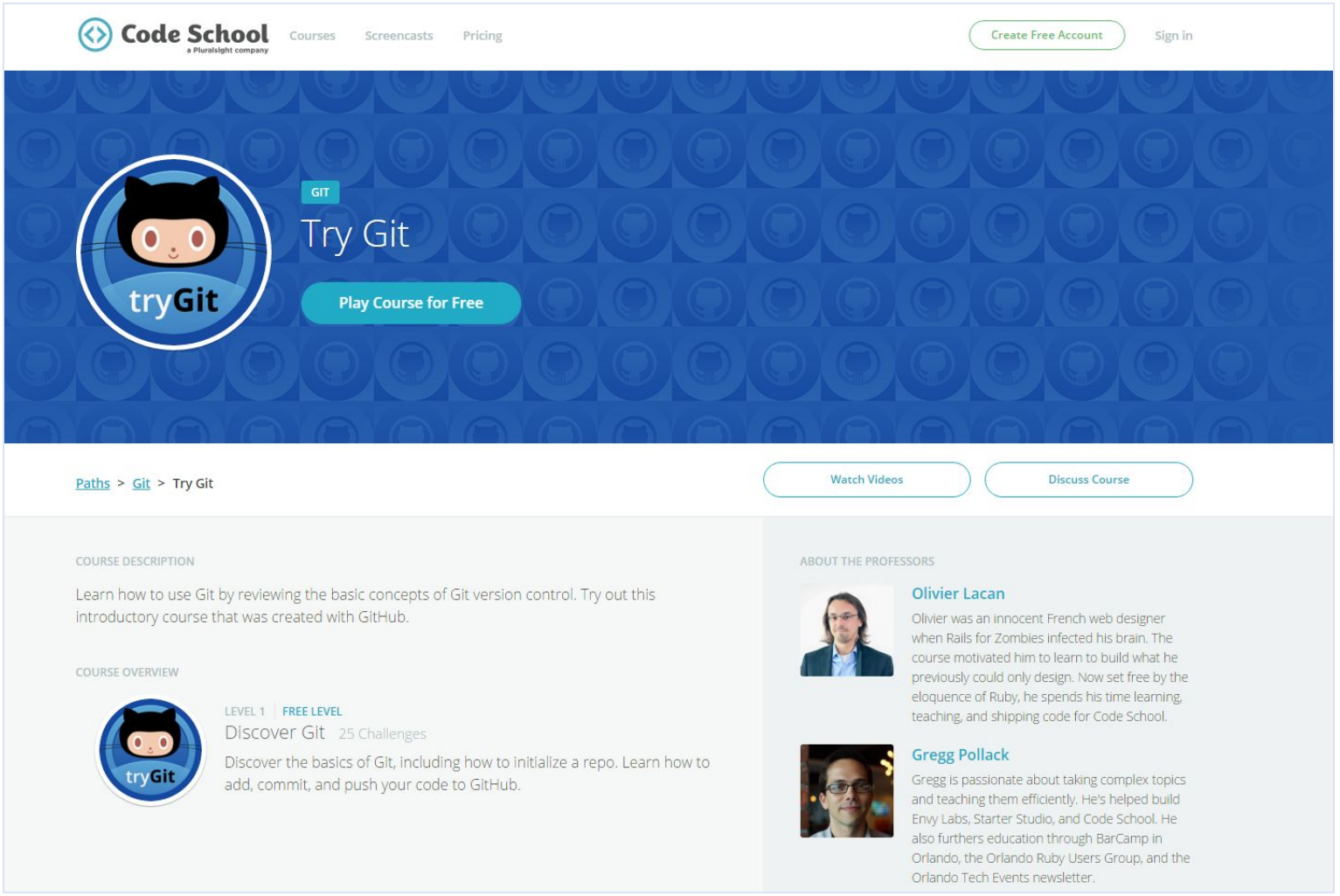
Step 1

Create a New Repository in GitHub.com



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

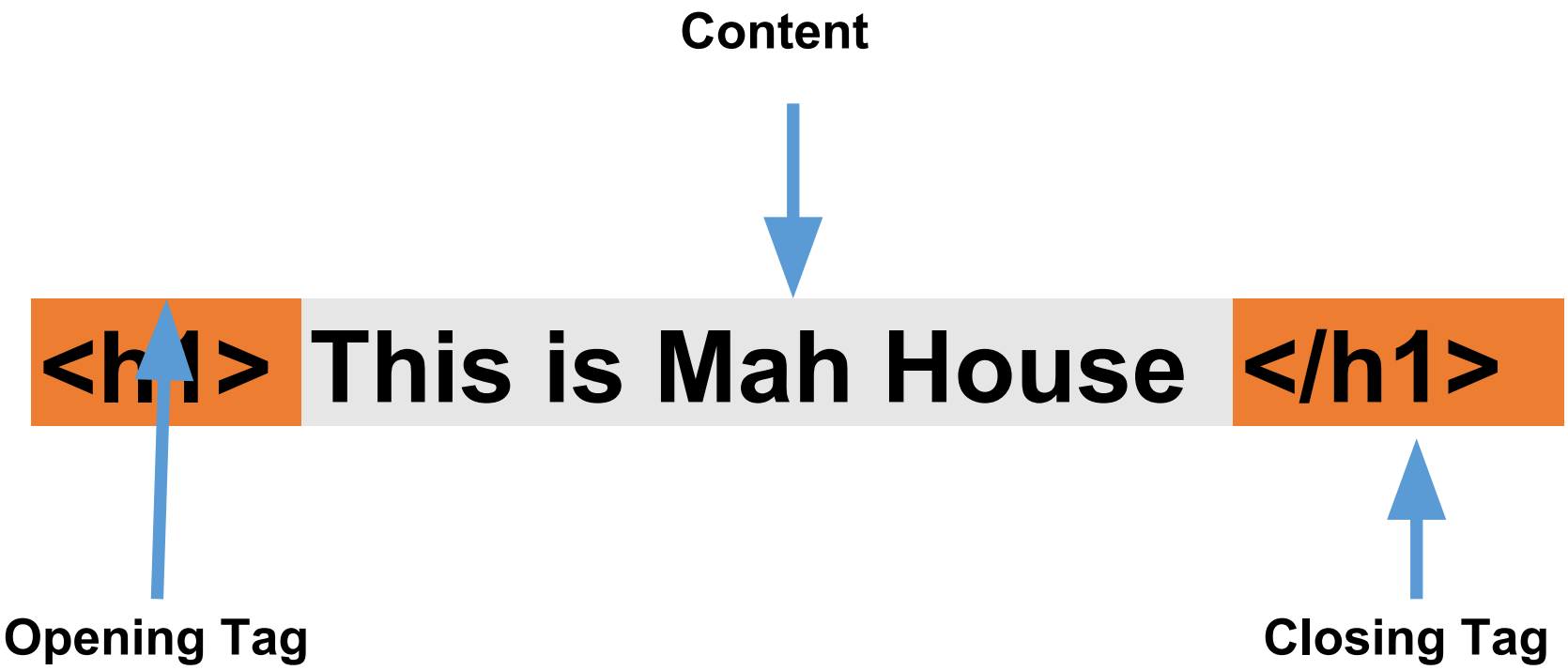
If You're Still Lost… Here's a (Free) Course



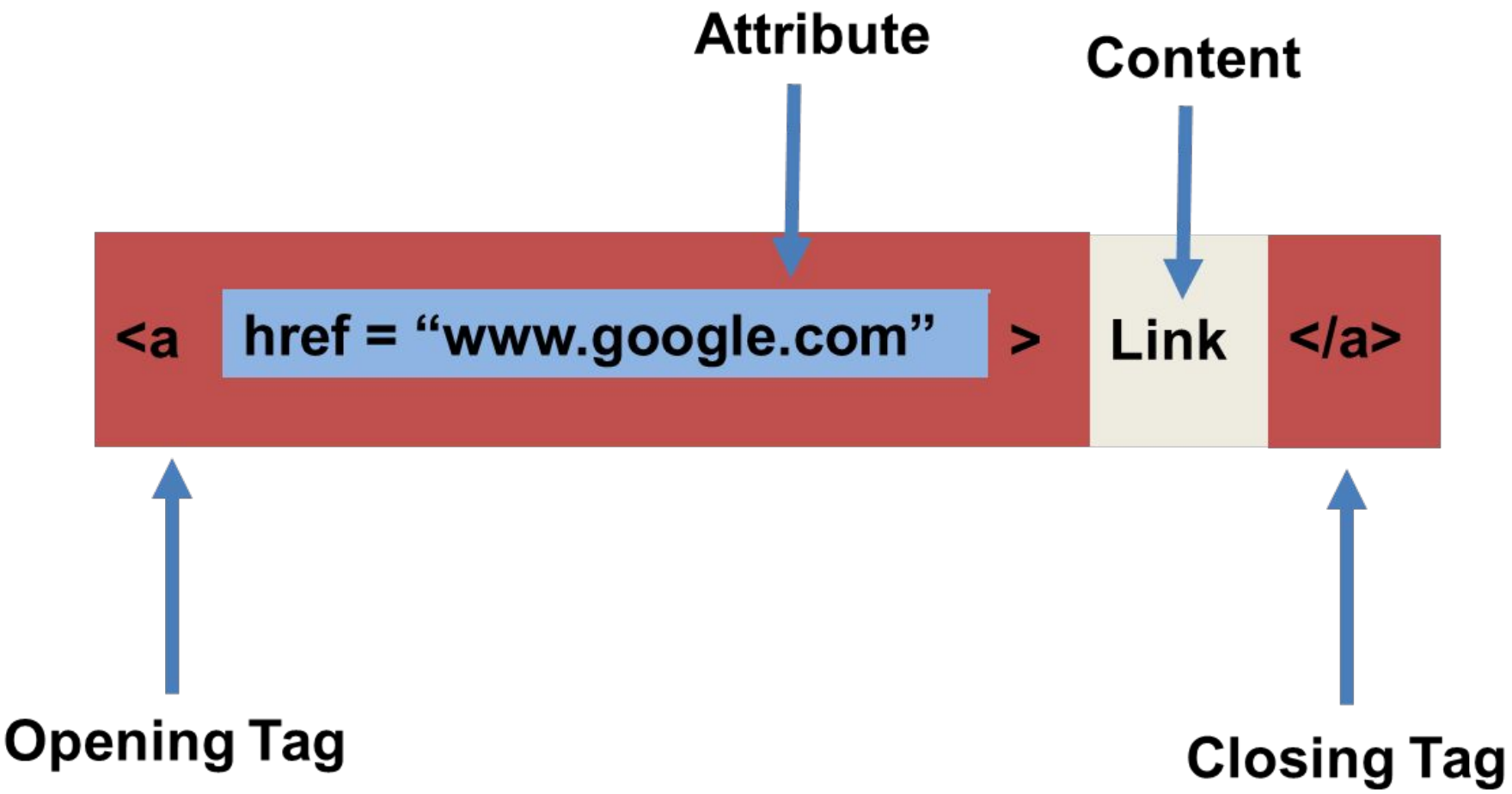
https://www.codeschool.com/courses/try-git

HTML Round 2

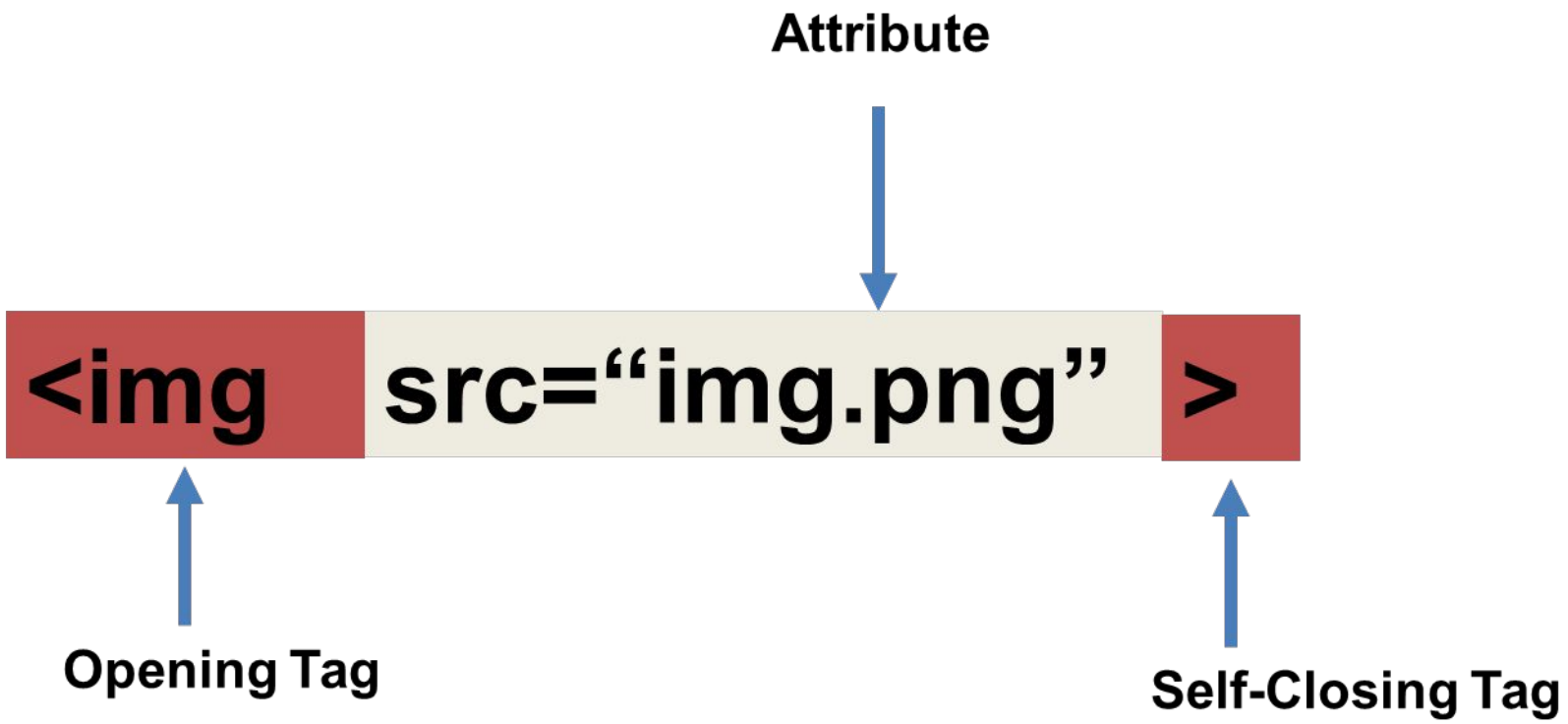
HTML Syntax (Basic)



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
- ** ** - Logical Container ***
- ** ** - Wraps individual Paragraphs

Others:

- **** (bold), **** (emphasis)
- **<img>** (images)**, <a href>** (links)**, ** (list items) **, <title>** (title), **
** (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

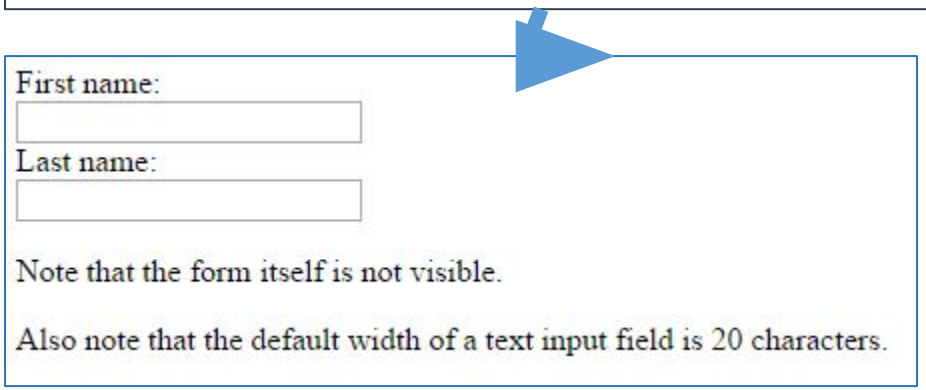
HTML for Forms

Common UI (User Interface) Form Elements:

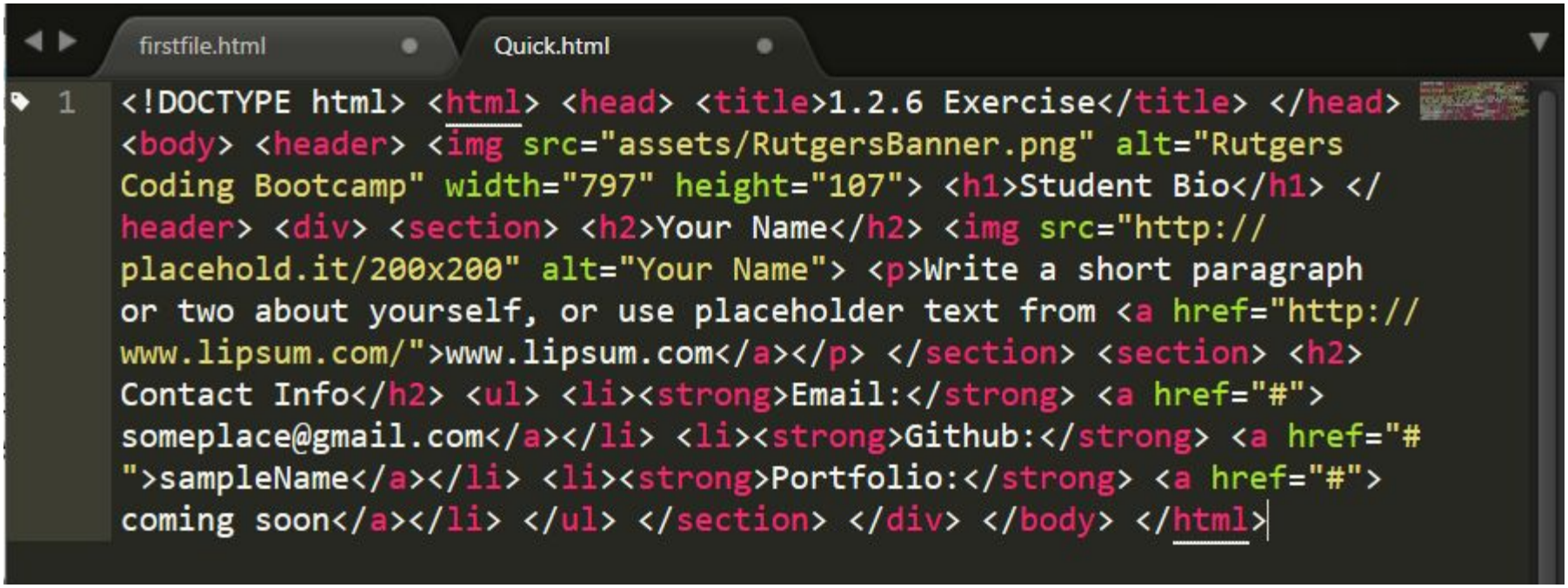
- **<form>** - Creates a form section in HTML
- **<input>** - Input boxes
- **<label>** - Labels for boxes
- **<button>** - Button
- **<textarea>** - Large textbox

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>
```



On Ugly 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, sed do eiusmod tempor
        incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
        exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.</p>
    <img src="https://vignette.wikia.nocookie.net/Bill_Murray.jpeg/" alt="bill murray" width="25%" >

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

Hella Boring…

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?

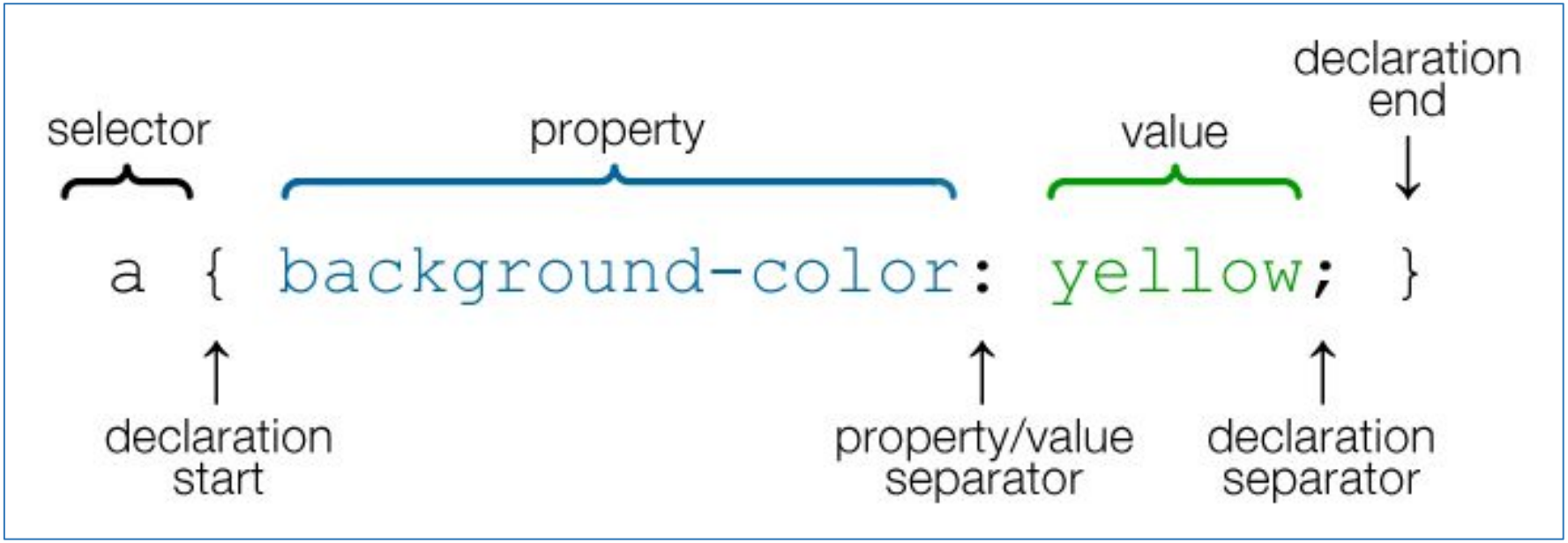
Menu Links

- 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.



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>
```

Example (CSS):

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

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, centers, or tops.

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

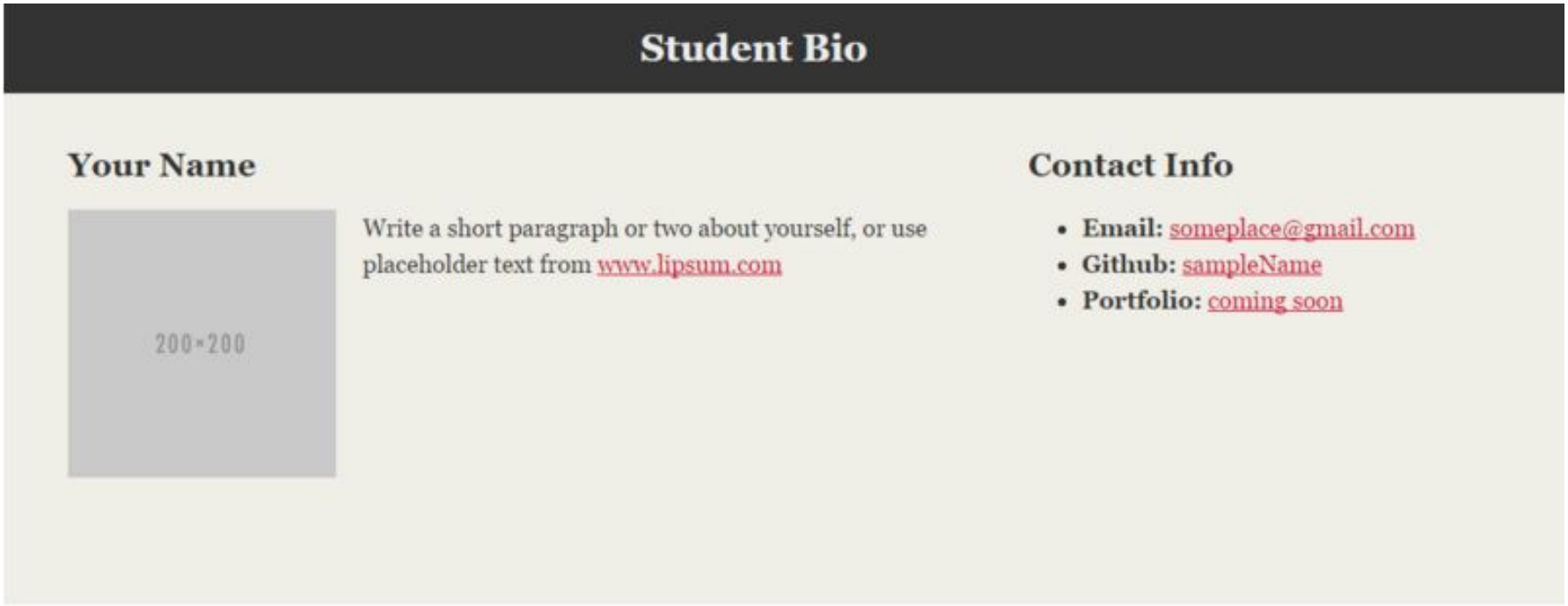
(quickexample_internalcss.html | 2-BasicCSS)

Assignment

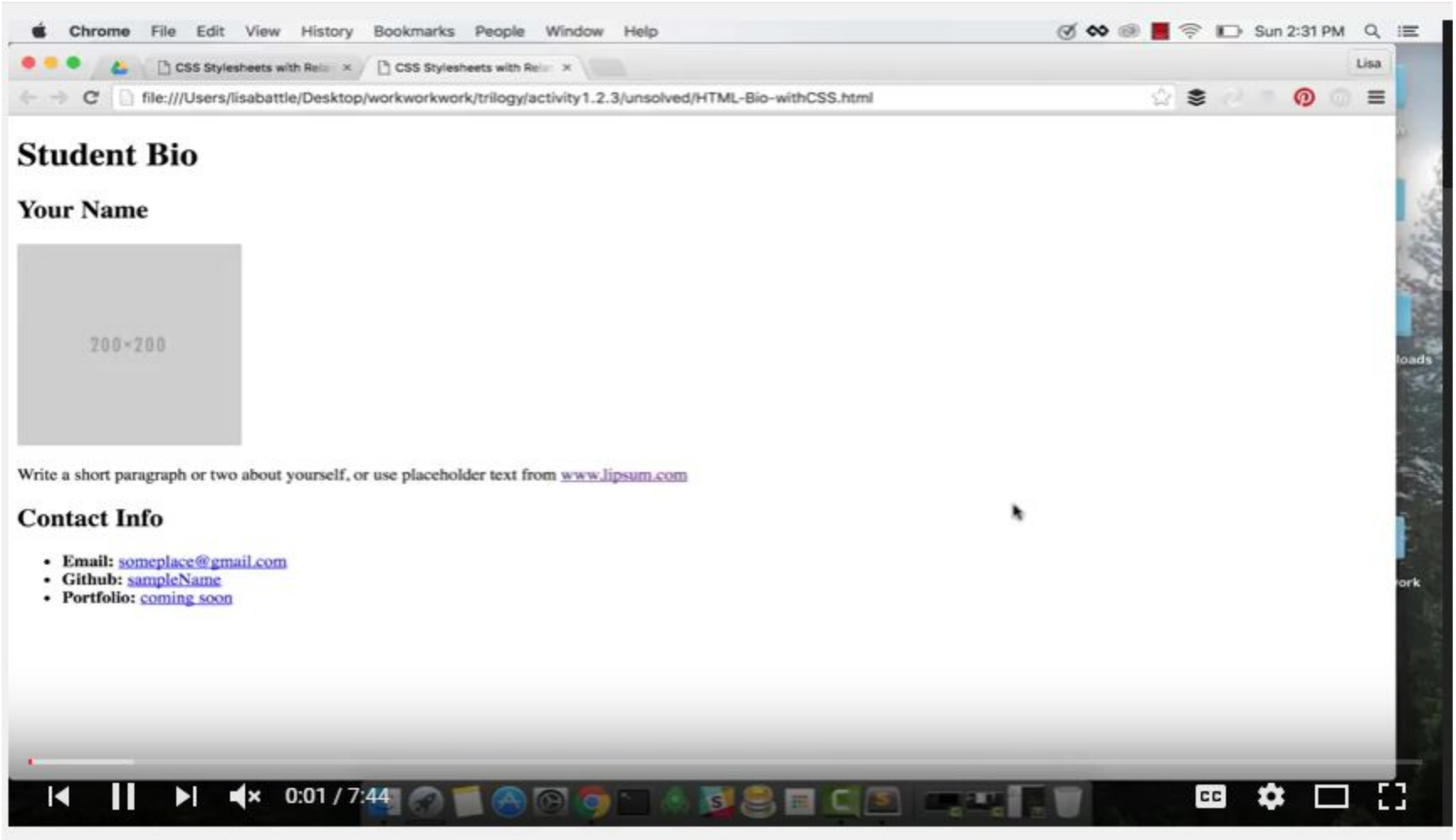
In this activity, you'll upgrade your previous HTML bio-page using CSS style rules. Once you're done, commit and push up your changes to GitHub.

We'll send you additional instructions via Slack.

> YOUR TURN!


Student Bio
Your Name
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

Video Walkthrough!!



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