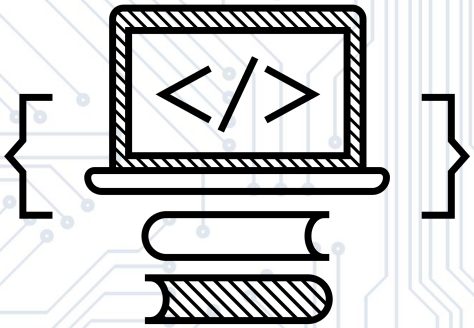
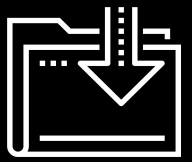


Git’n Pro with HTML/CSS

Web Development Boot Camp

Lesson 1.2



Admin Items

Admin items

- File organization: Dan’s guide in #03-resources
- Still no access to the class repo on GitLab? Slack Ben.
- Breakout rooms

How do I do this again?

How to Get Help

01 Practice, practice, practice: work individually or in groups.

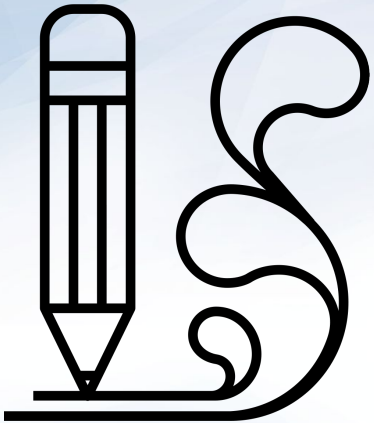
02 Review in-class material (activities and slides).

03 Watch the class videos again.

04 Attend office hours, which are held 45 minutes before and 30 minutes after class.

05 Attend one-on-one sessions with your Student Success Manager (SSM) (to be announced by your SSM).

06 Contact your Student Success Manager anytime!



Homework Assignment: **Prework**

Due Date:
Finish by Saturday August 29th

Today’s Class!

Today's Objectives

Today we will:

01 Understand the importance of Git version control and how to use it.

02 Create GitHub repositories, push up code, and share with the class.

03 Create more HTML documents.

04 Learn how to properly use basic HTML tags.

05 Apply basic CSS styling to HTML documents.

Know Thyself

If you are a beginner to HTML/CSS and coding, your objectives are to:

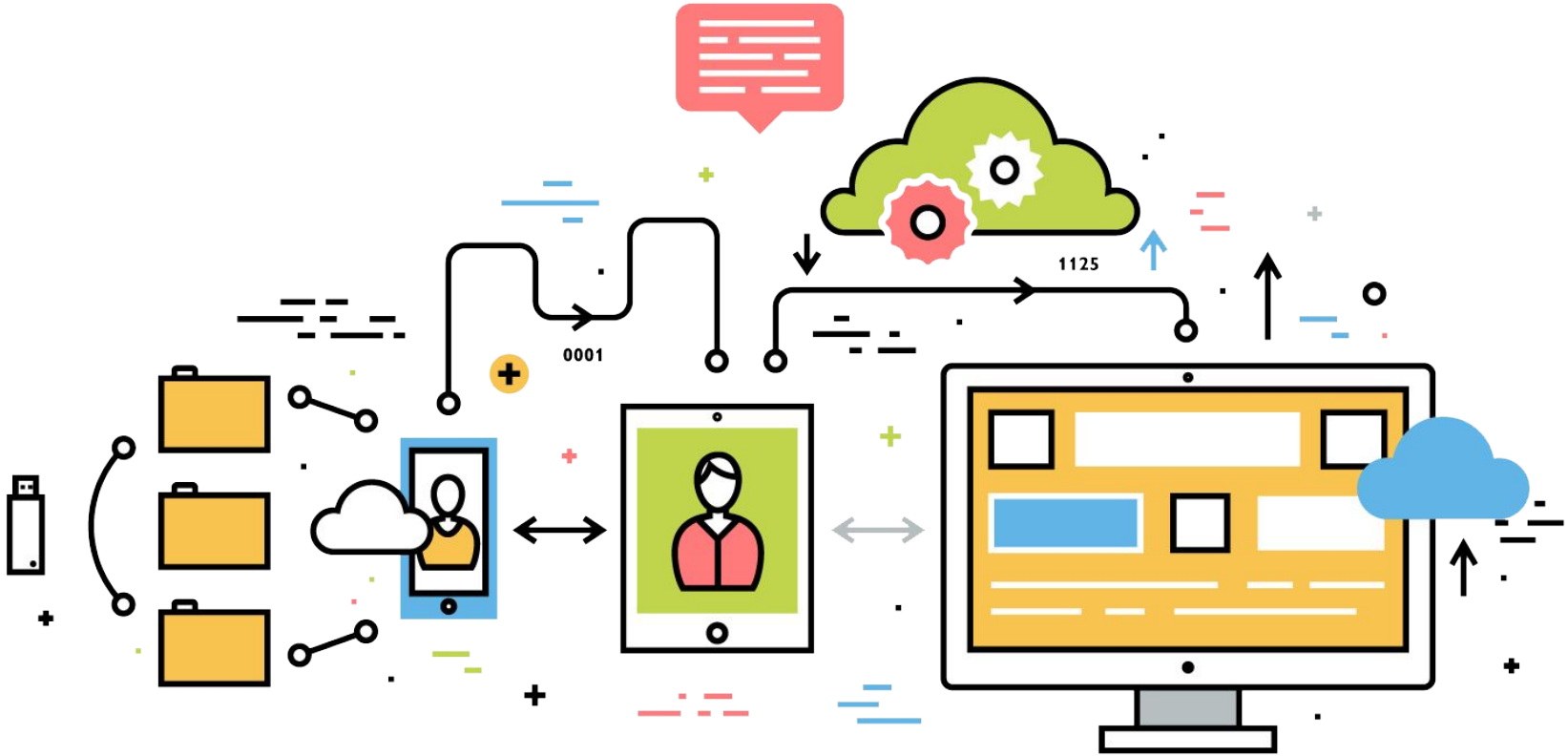
- Continue to get comfortable with HTML.
- Be able to write a complete, basic HTML document (like in the last class).
- Understand the function of CSS and how it works with HTML.
- Be able to use Git and GitHub to upload code.

If you've had past exposure to HTML/CSS and coding and felt comfortable with the last lesson, your objectives are to:

- Aim to build up your skills.
- Clear up any questions or confusion you have 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.

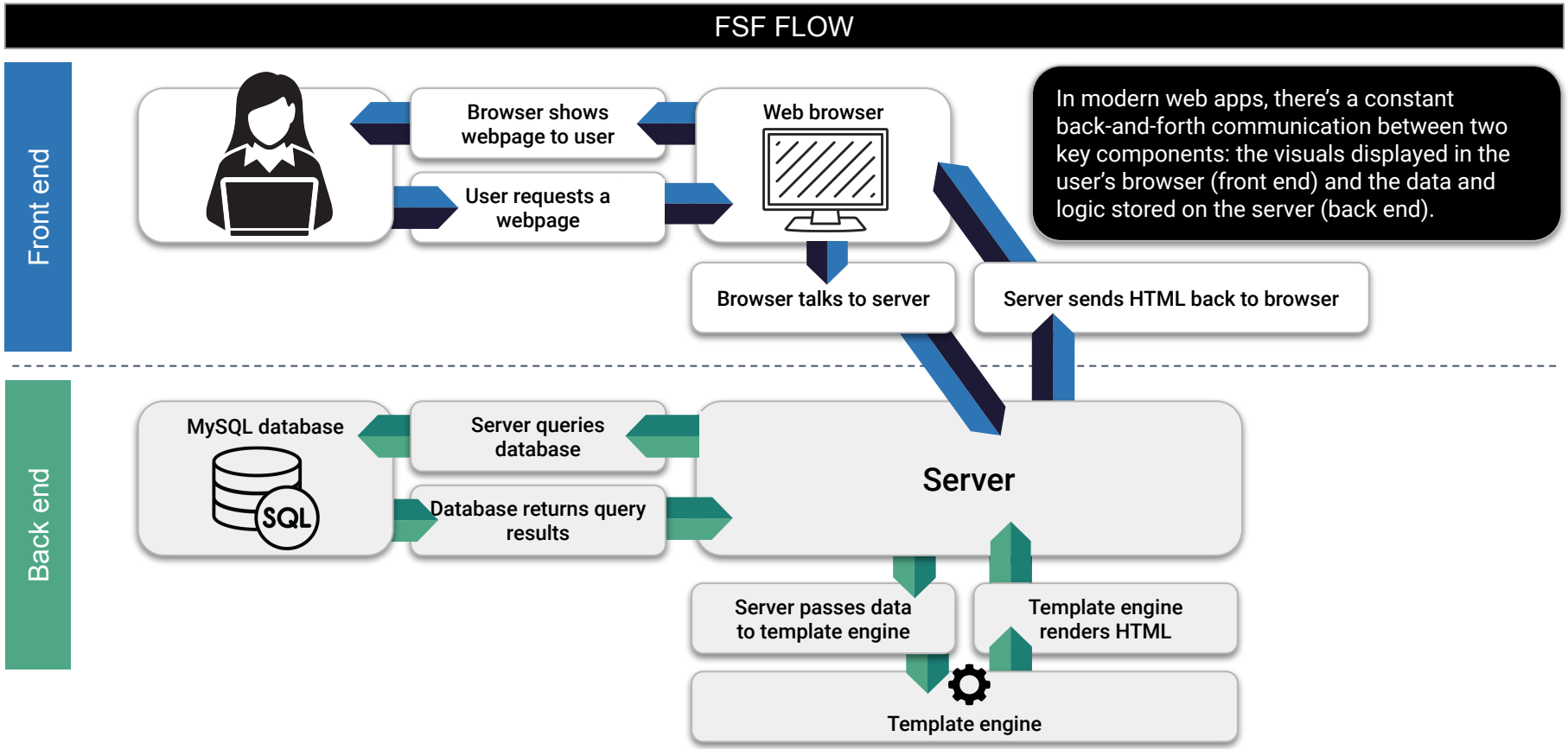
The Modern Web

What Exactly Is Full-Stack Development?

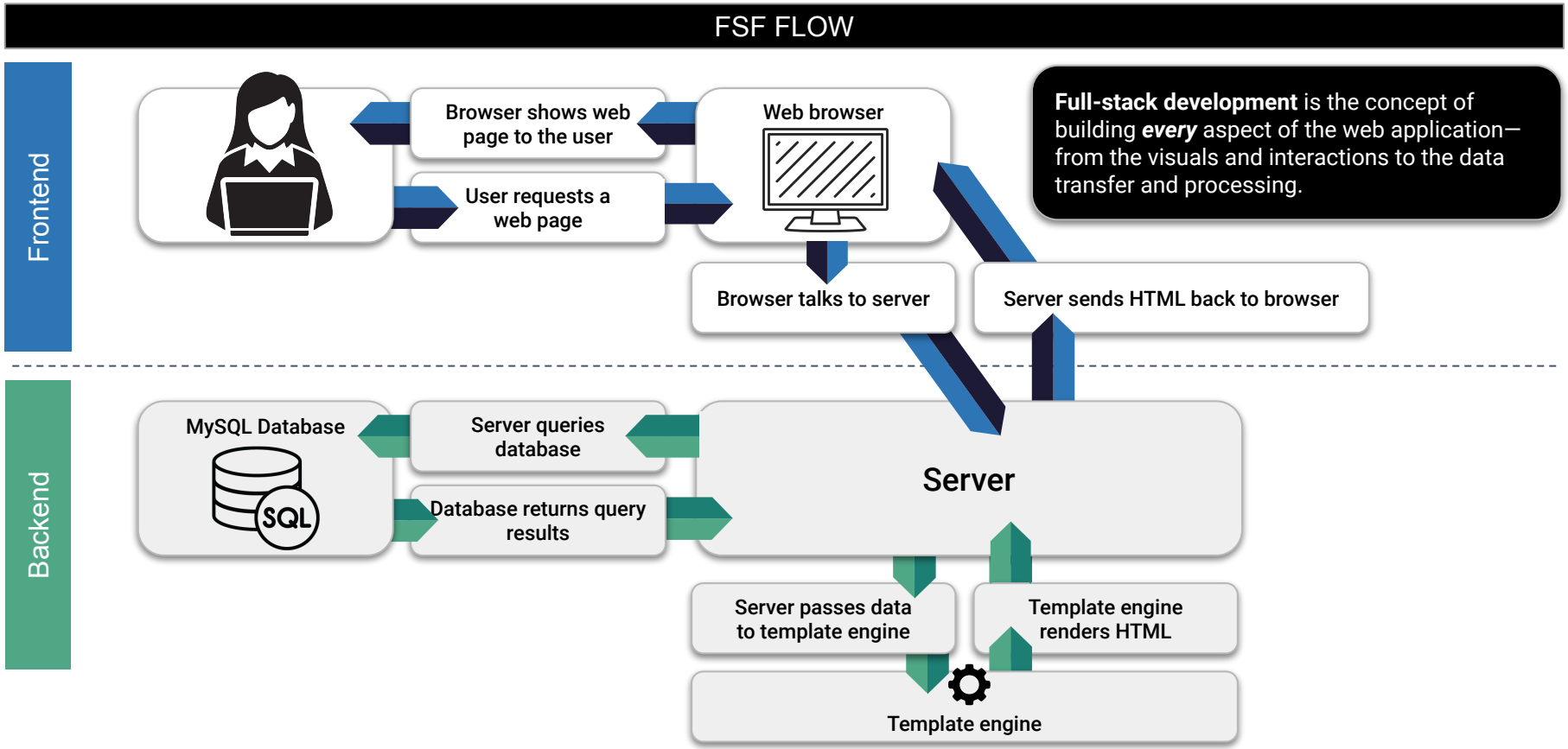


The “Magic” of YouTube

Full-Stack Development



Full-Stack Development



Full-Stack Development

The Browser	Dev Tools	Server Side
HTML	Heroku	Templating engines
CSS	Git	Sessions
JavaScript	GitHub	Writing tests
jQuery	**Databases**	Node.js
Bootstrap	MySQL	Express.js
SEO	MongoDB	Creating APIs
API Interaction		MVC
APIs (Consuming)		User authentication
JSON		ORM (Object-relational mapping)
AJAX		**CS Fundamentals**
Real-time cloud database via Firebase		Algorithms
Cutting-Edge Development		Design patterns
React.js		

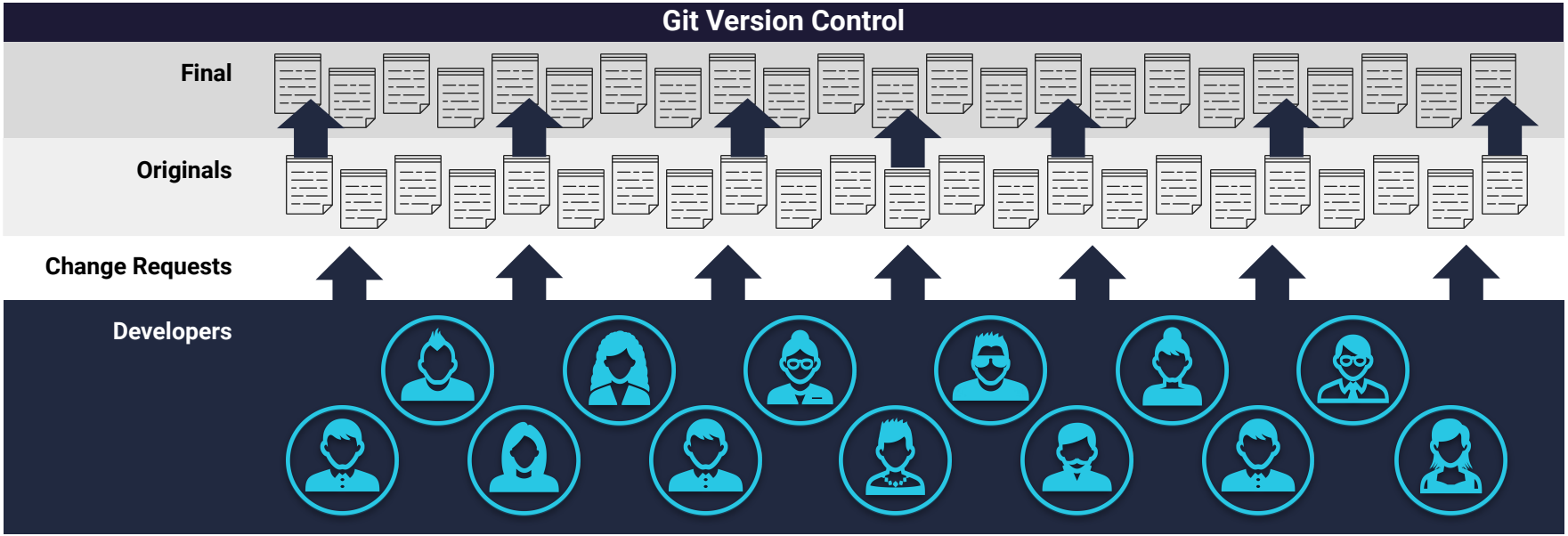
What Is Git?

Collaborative Coding

Modern web development is *highly* collaborative.

Teams are often extremely large and spread out across the country or world.

Apps are sometimes made up of hundreds or even thousands of files.



The Team's Task

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

Programming Team		
Maya Angelou	Anne Sexton	Gil Scott Heron
		

Maya Angelou & Gil Scott Heron Make Their Edits

Maya Angelou is programming away.

Maya Angelou's version

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

Gil Scott Heron is programming away.

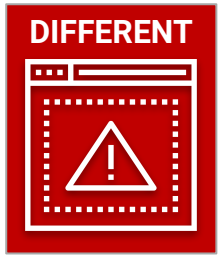
Gil Scott Heron's version

```
<ul>
    <li>Free Will</li>
    <li>Pieces of a Man</li>
    <li>The Revolution Will Not Be
        Televised</li>
</ul>
```

Different Solutions

Maya Angelou's version

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



Gil Scott Heron's version

```
<ul>
    <li>Free Will</li>
    <li>Pieces of a Man</li>
    <li>The Revolution Will Not Be Televised</li>
</ul>
```

Resolution

Maya Angelou's version

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

Gil Scott Heron's version

```
<ul>
    <li>Free Will</li>
    <li>Pieces of a Man</li>
    <li>The Revolution Will Not Be Televised</li>
</ul>
```

Let's settle on this:

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

Anne Sexton Writes Her Own Version

Anne Sexton's version

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

Anne Sexton Overwrites the Work of Her Teammates

Delete. Delete.
Delete. Delete.
Delete. Delete.

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

```
<ul>
    <li>The Double Image</li>
    <li>45 Mercy Street</li>
    <li>The Road Back</li>
</ul>
```

The Group Project

Lesson: You should use version control because it helps you manage multiple developers working on a single codebase.

"Today we fret and pull on wheels, ignore our regular loss of time…" Or maybe we should just use Git.

Git Version Control

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

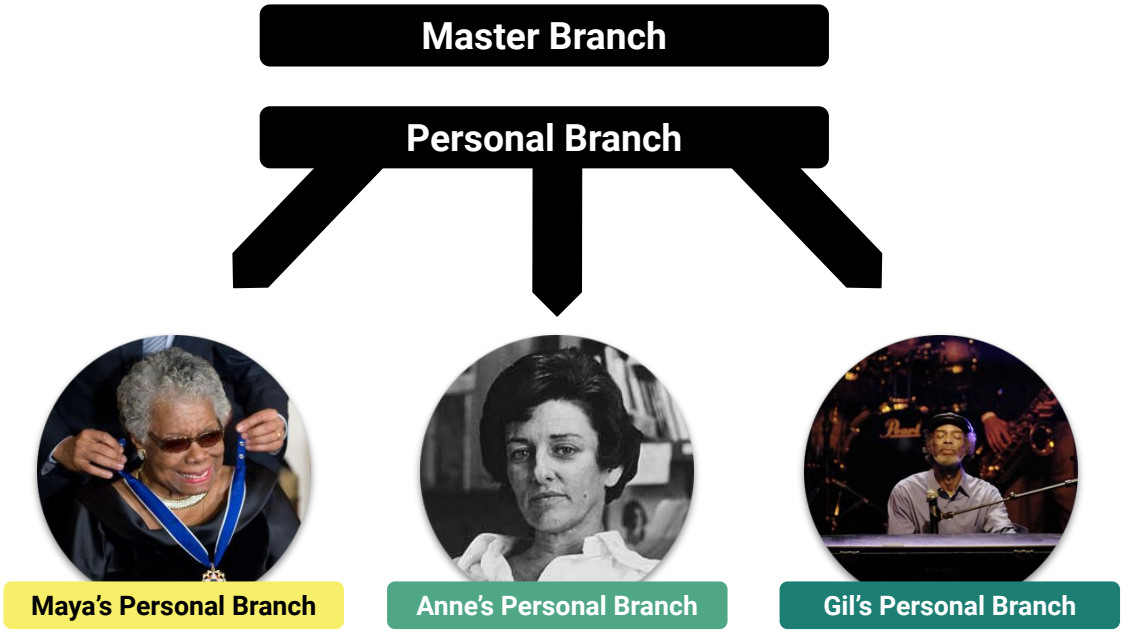
The Benefits of Git 

01 A process for resolving conflicts in code

02 Version history

The Group Project

Branch = personal copy



The Team Goes to Work

Maya Angelou's version

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

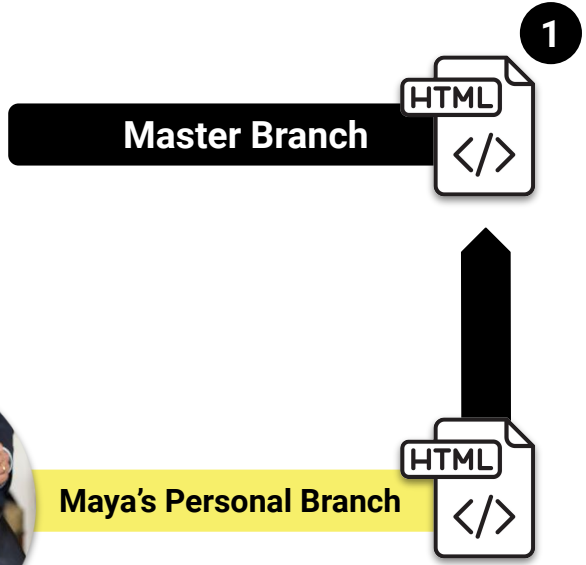
Gil Scott Heron's version

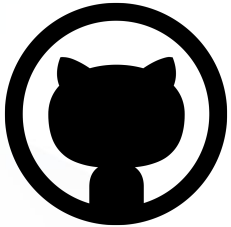
```
<ul>
      <li>Free Will</li>
      <li>Pieces of a Man</li>
      <li>The Revolution Will Not Be Televised</li>
</ul>
```

Maya Angelou Pushes First

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

No code conflicts

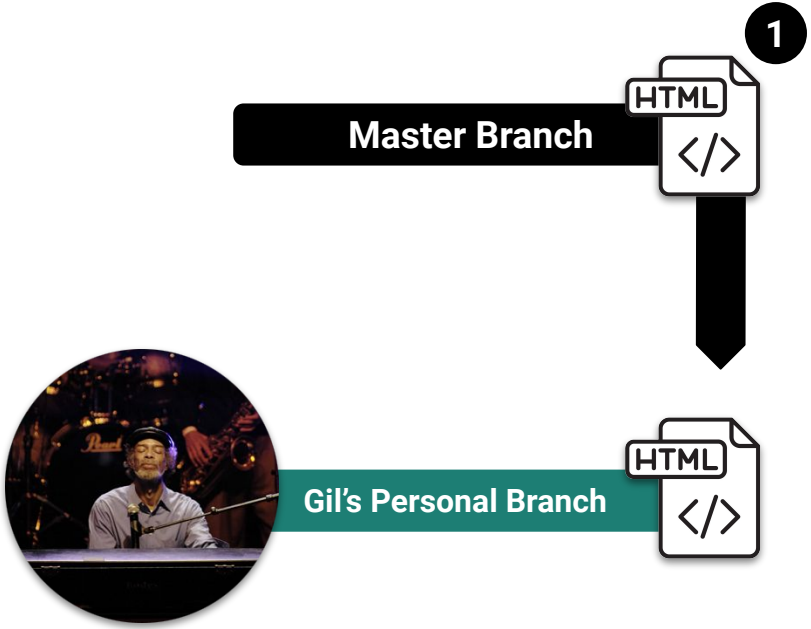




Rule: Pull first, and then push your changes.

Gil Scott Heron's Edits Are Ready

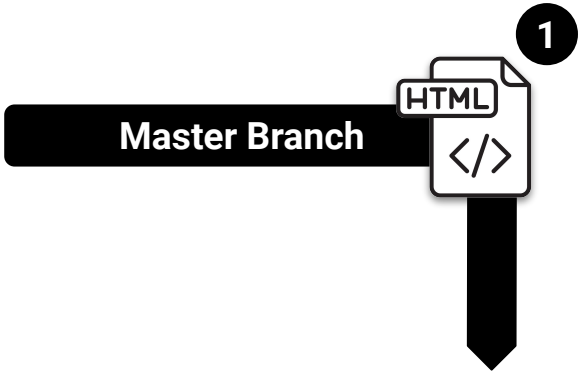
Rule: Pull first, and then push your changes.



Gil Scott Heron Conflicts with the Master Branch

Git sees a conflict.

```
<ul>
<<<<<<<<<< HEAD
      <li>Free Will</li>
      <li>Pieces of a Man</li>
      <li>The Revolution Will Not Be Televised</li>
==========
      <li>On the Pulse of Morning</li>
      <li>I Know Why the Caged Bird Sings</li>
      <li>And Still I Rise</li>
>>>>>>>>>> master
</ul>
```



Gil Resolves

```
<ul>
<<<<<<<<<< HEAD
      <li>Free Will</li>
      <li>Pieces of a Man</li>
      <li>The Revolution Will Not Be Televised</li>
==========
      <li>On the Pulse of Morning</li>
      <li>I Know Why the Caged Bird Sings</li>
      <li>And Still I Rise</li>
>>>>>>>>>> master
</ul>
```

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

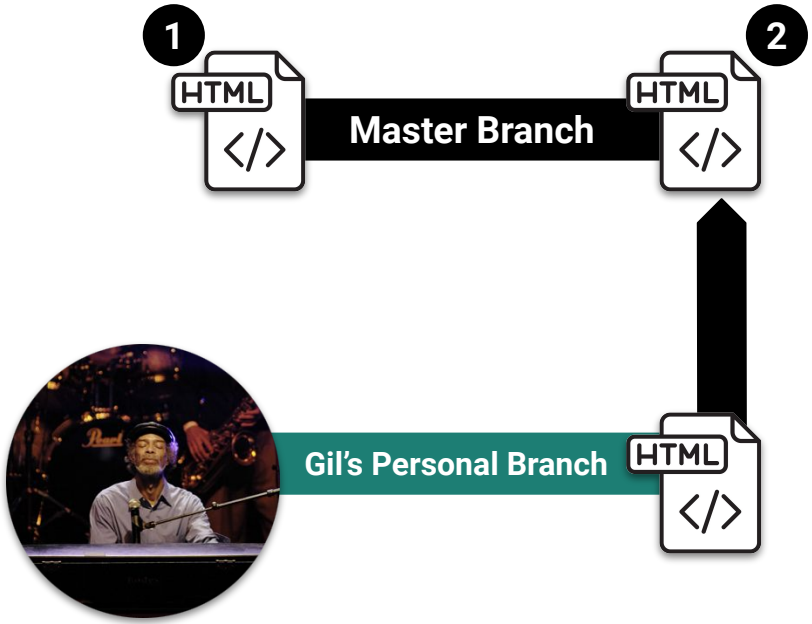
Gil's Personal Branch

Gil Scott Heron Fixes and Pushes

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

No code conflicts

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



Anne Sexton Starts Her Work

Rule: Pull first, and then push your changes.

look into my face

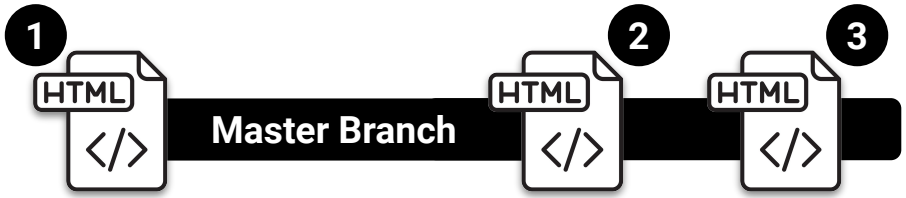
and you will know that crimes dropped upon me

as from a high building…

…by which I mean, I broke the rules.

Anne Sexton Pushes

Anne Sexton pushes (uploads) her revision to the main branch, but she doesn't pull first. Because she did not pull first, she sees no conflicts in the code (and doesn't get Gil's work!). **This is not what we want.**




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

If Anne Had Made a Pull First...

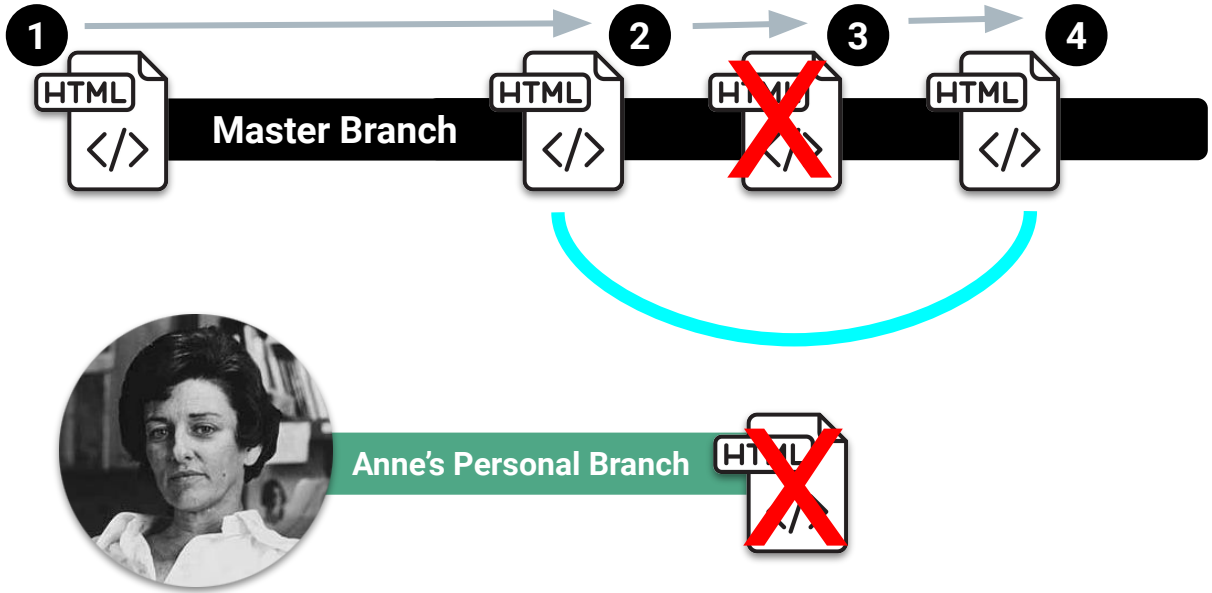
Conflict!

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

The **overwritten** work is discovered!

Roll Back

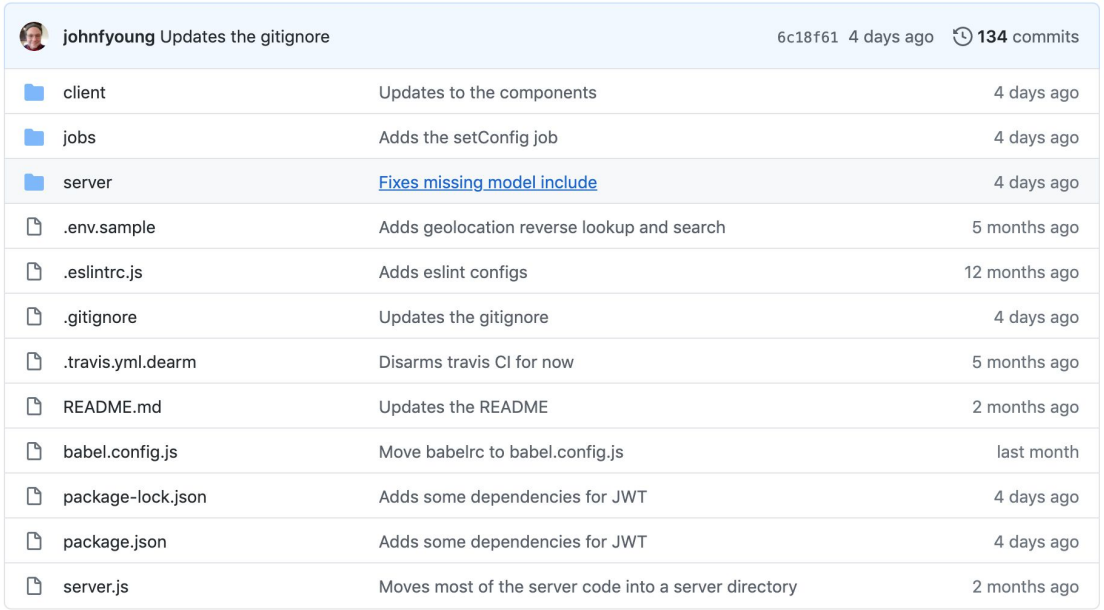
Maya **rolls back** the code to an earlier version.



Git is change management:
a computer file system
with snapshots over time.

Git: what is a commit?

A commit is a bundle of changes to your project files



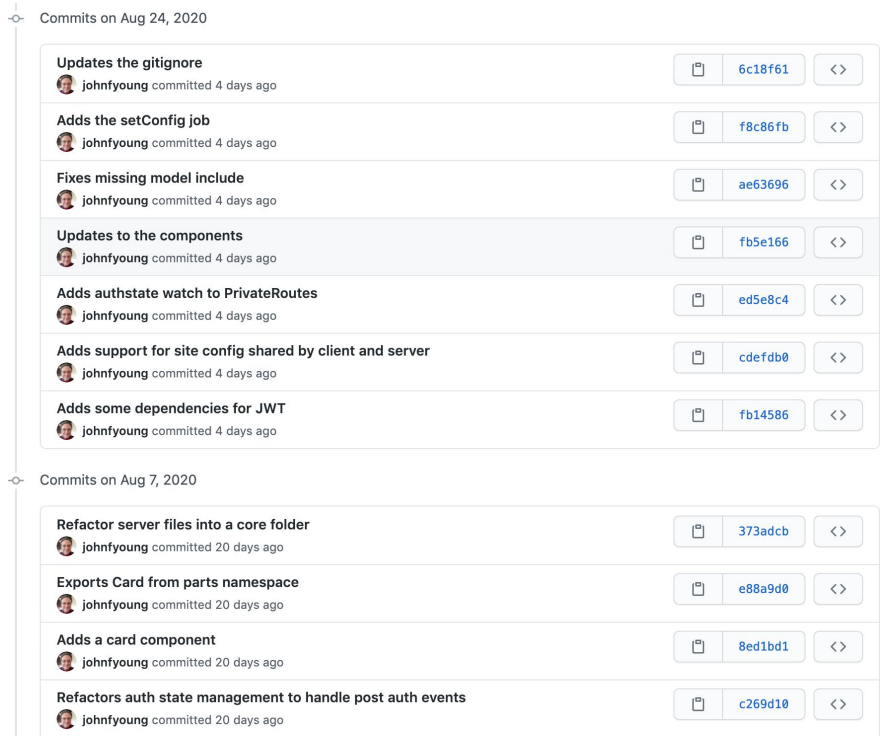
johnfyoung Updates the gitignore	6c18f61 4 days ago	134 commits
client	Updates to the components	4 days ago
jobs	Adds the setConfig job	4 days ago
server	Fixes missing model include	4 days ago
.env.sample	Adds geolocation reverse lookup and search	5 months ago
.eslintrc.js	Adds eslint configs	12 months ago
.gitignore	Updates the gitignore	4 days ago
.travis.yml.dearm	Disarms travis CI for now	5 months ago
README.md	Updates the README	2 months ago
babel.config.js	Move babelrc to babel.config.js	last month
package-lock.json	Adds some dependencies for JWT	4 days ago
package.json	Adds some dependencies for JWT	4 days ago
server.js	Moves most of the server code into a server directory	2 months ago

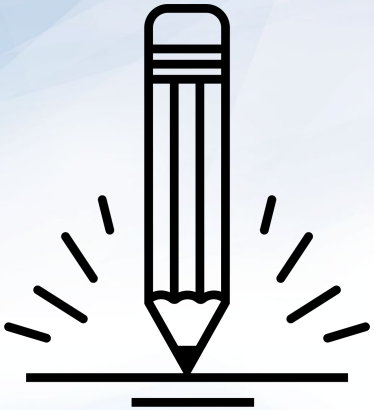
Important:

- Making a commit marks a snapshot of your project.
- If needed (i.e., you've made a mistake and would like to rewind time), you can roll your project back to that point

Git: what is a commit?

Commits mark rollback points





Activity:
Explain Git (Version Control)

Suggested Time:
3 Minutes

Activity: Explain Git (Version Control)

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

The concept of version control

Then, the other should explain:

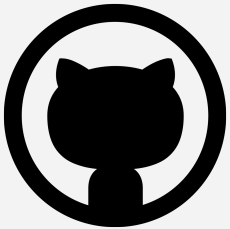
Two key advantages of using a version control system

Suggested Time: 3 Minutes

What Is GitHub?

01

GitHub is a web-based hosting service to store code online.



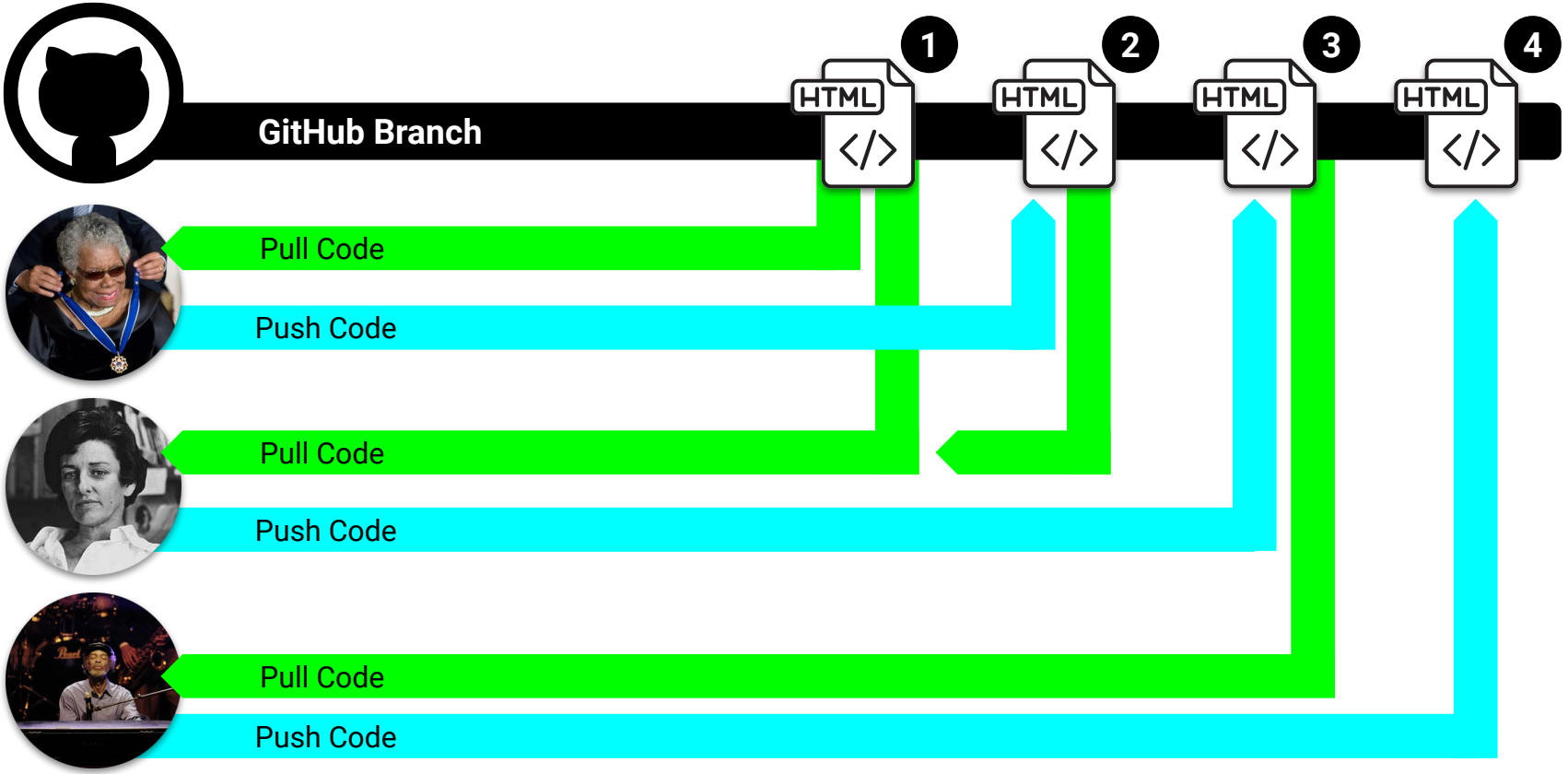
02

It allows developers to pull (download) code or push (upload) code to the same repository (directory).

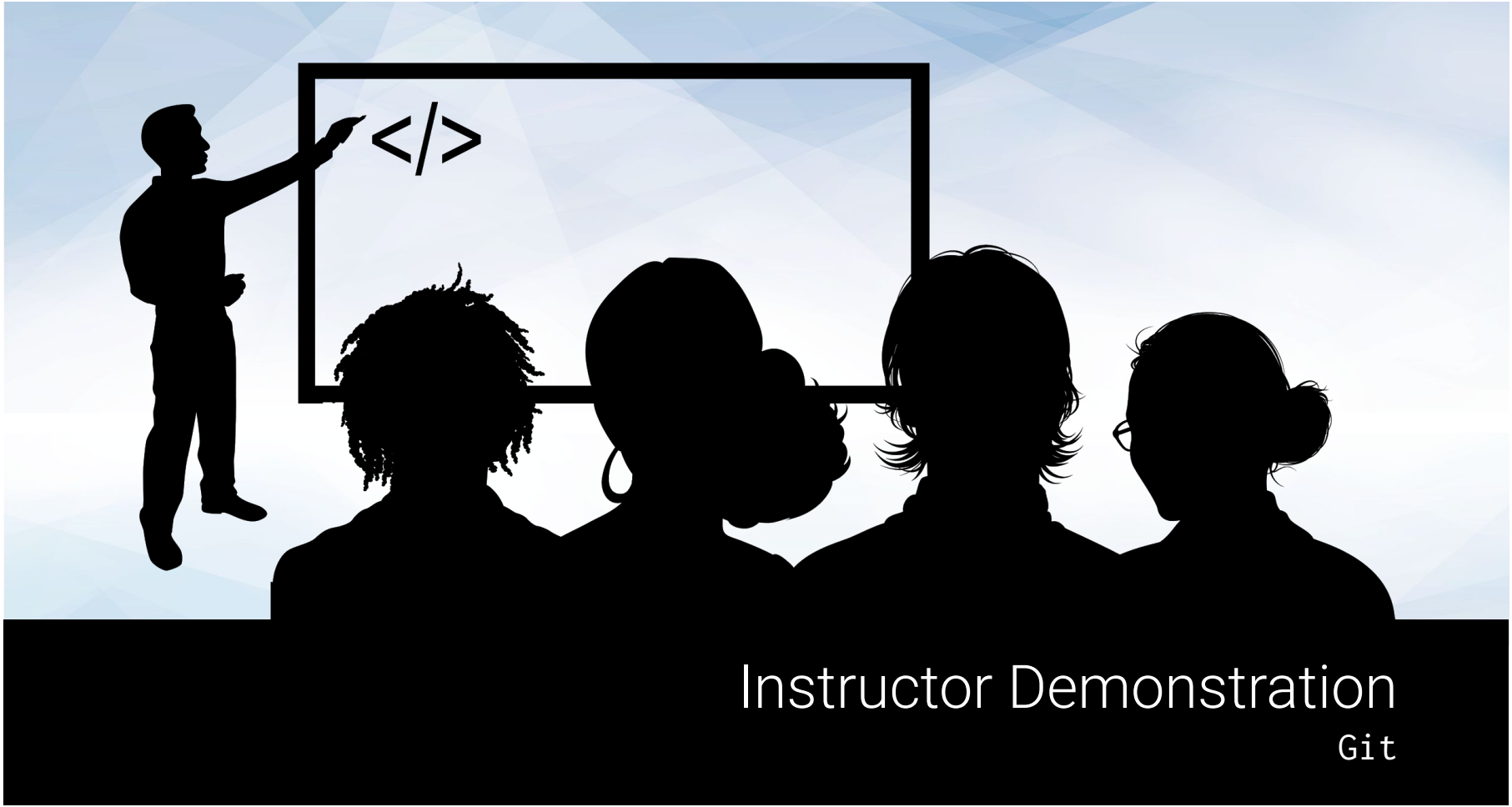
03

It also allows developers to view histories of code changes and track issues.

Pushing and Pulling to GitHub



Get Started with Git



Instructor Demonstration

Git

Basic Git Commands

These are the six basic Git commands to get started:

01 `git clone`

02 `git add`

03 `git commit`

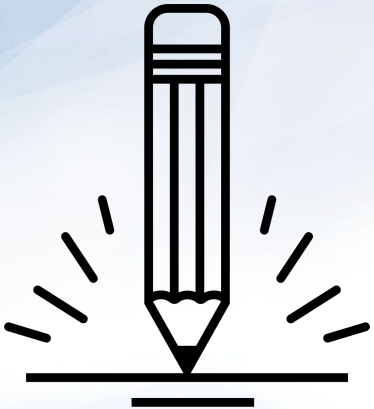
04 `git push`

05 `git pull`

06 `git status`

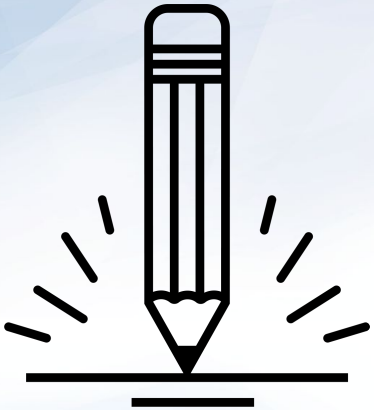
Basic Git Commands

`git clone`	Copies an entire repo (to begin)
`git add`	Adds a changed file for inclusion in a commit
`git commit`	Marks a set of change to the local repo
`git push`	Sends changes to hosting service
`git pull`	Downloads freshest version of repo
`git status`	See what has changed in your working copy



Activity:
Clone the class repo

Suggested Time:
2 minutes



Activity:

Git Add, Commit, Push

Suggested Time:
20 minutes

Activity: `Git Add, Commit, Push`

Using GitHub and the command line:

1. Create a new **public GitHub repository** and name it whatever you like. Be sure to check the box to initialize this repository with a README.
2. **Clone** the repo to your local directory.
3. Create an **HTML file** inside the local directory.
4. **Add, commit**, and **push** the code to GitHub.

Bonus:

1. Find a partner in class, and **fork** *their* repository to your own GitHub account. Clone this forked repository to your local directory.
2. **Add, commit,** and **push** the code back to your forked copy.
3. Submit a **pull request** to send your changes to your partner's repo.

Suggested Time: 20 Minutes

Still a Bit Lost? Don’t Worry!

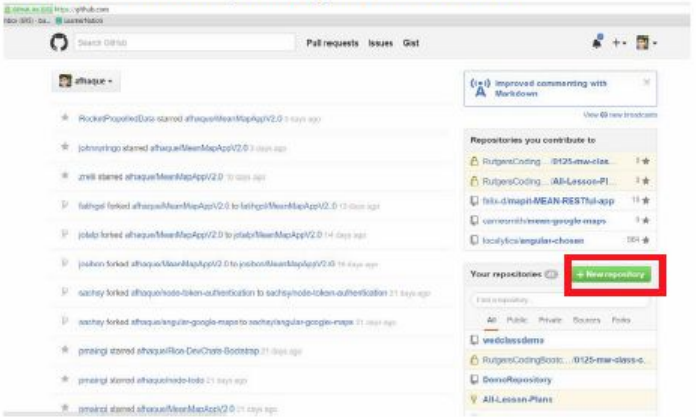
Follow this handy guide!

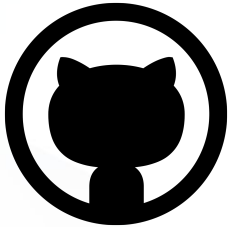
Practice a few times on your own before the next class.

Steps to Uploading Your Code to GitHub

Step 1

Create a New Repository in GitHub.com





If you’re still lost, here’s a (free) course on how to use GitHub:

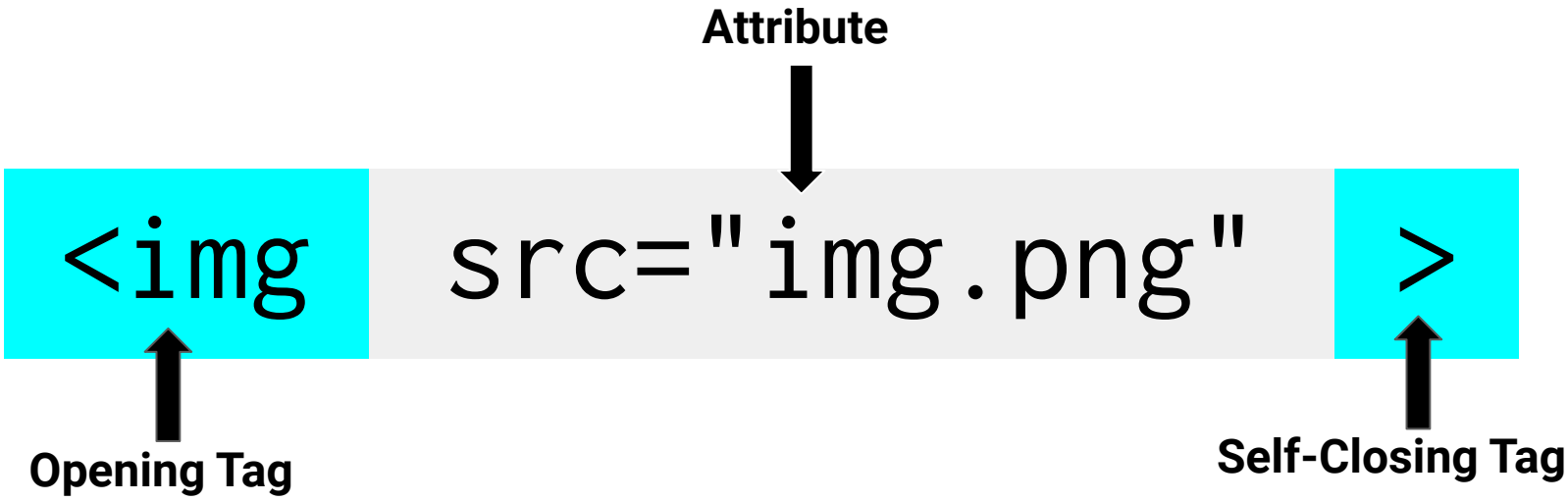
Get Started with Github

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
` `	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.

`<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

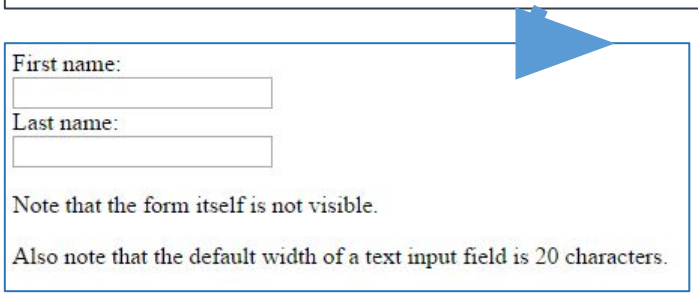
HTML for Forms

Common UI (user interface) form elements:

`<form>`	Creates a form section in HTML
`<input>`	Input boxes
`<label>`	Labels for boxes
`<button>`	Button
`<textarea>`	Large text box

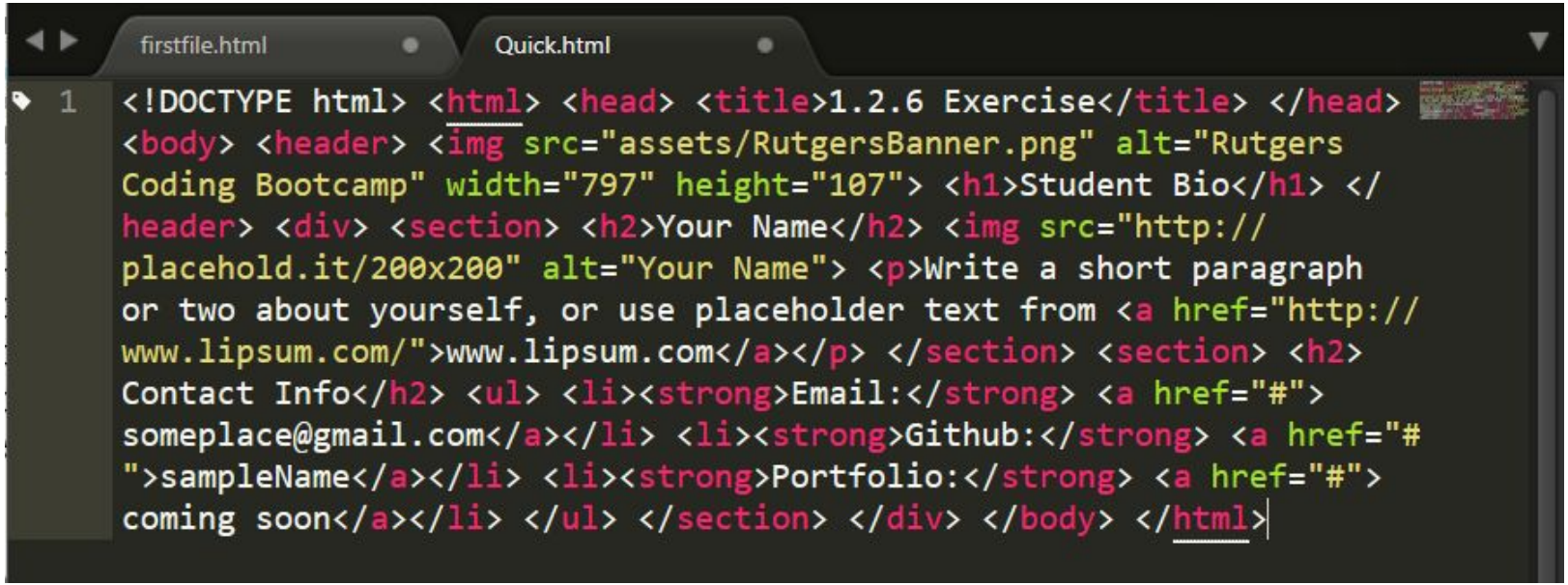
HTML for Forms

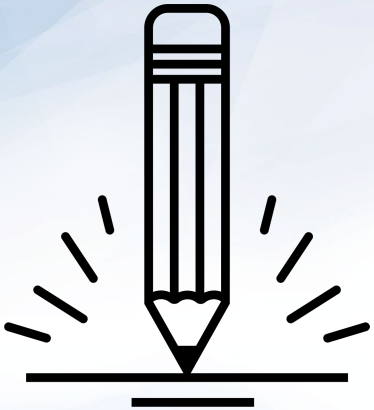
```
<!DOCTYPE html>
<html>
<body>

<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>
```



Ugly HTML

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





Activity: Basic Student Bio

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 will be sent via Slack)

Suggested Time:
20 minutes

Activity: Basic Student Bio

Student Bio

Your Name

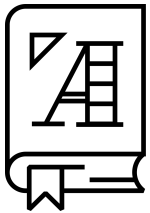
Write a short paragraph or two about yourself, or use placeholder text from www.lipsum.com.

Contact Info

- **Email:** someplace@gmail.com
- **GitHub:** Sample Name
- **Portfolio:** Coming Soon

CSS Stylin’

HTML and CSS Definitions



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 aesthetics of a webpage, including layouts, colors, fonts, and content placement. (JavaScript is the language that deals with logic, animation, etc.)

HTML/CSS Analogy

HTML Alone	HTML and CSS
Like writing papers in Notepad	Like writing papers in Microsoft Word
Can only write unformatted text	Can format text, page layout, alignment, and more 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 Awesome Header

Lorem ipsum dolor sit amet, consectetur adipiscing 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.

Menu Links

- **Google**
- **Facebook**
- **Twitter**

Basic HTML Page: Result

Awesome Header

Smaller Awesome Header

Even Smaller Awesome Header

Lorem ipsum dolor sit amet, consectetur adipiscing 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.

Boring

Menu Links

- Google
- Facebook
- Twitter

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 Awesome Header

Lorem ipsum dolor sit amet, consectetur adipiscing 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.

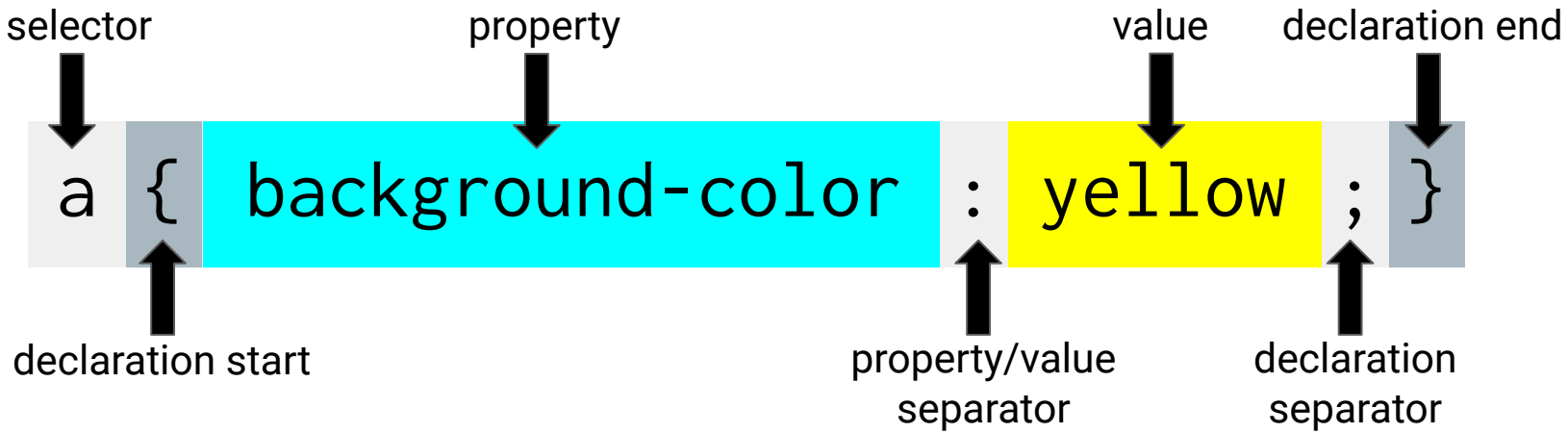
Menu Links

- **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 following example, the header would become 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)	Example (CSS)

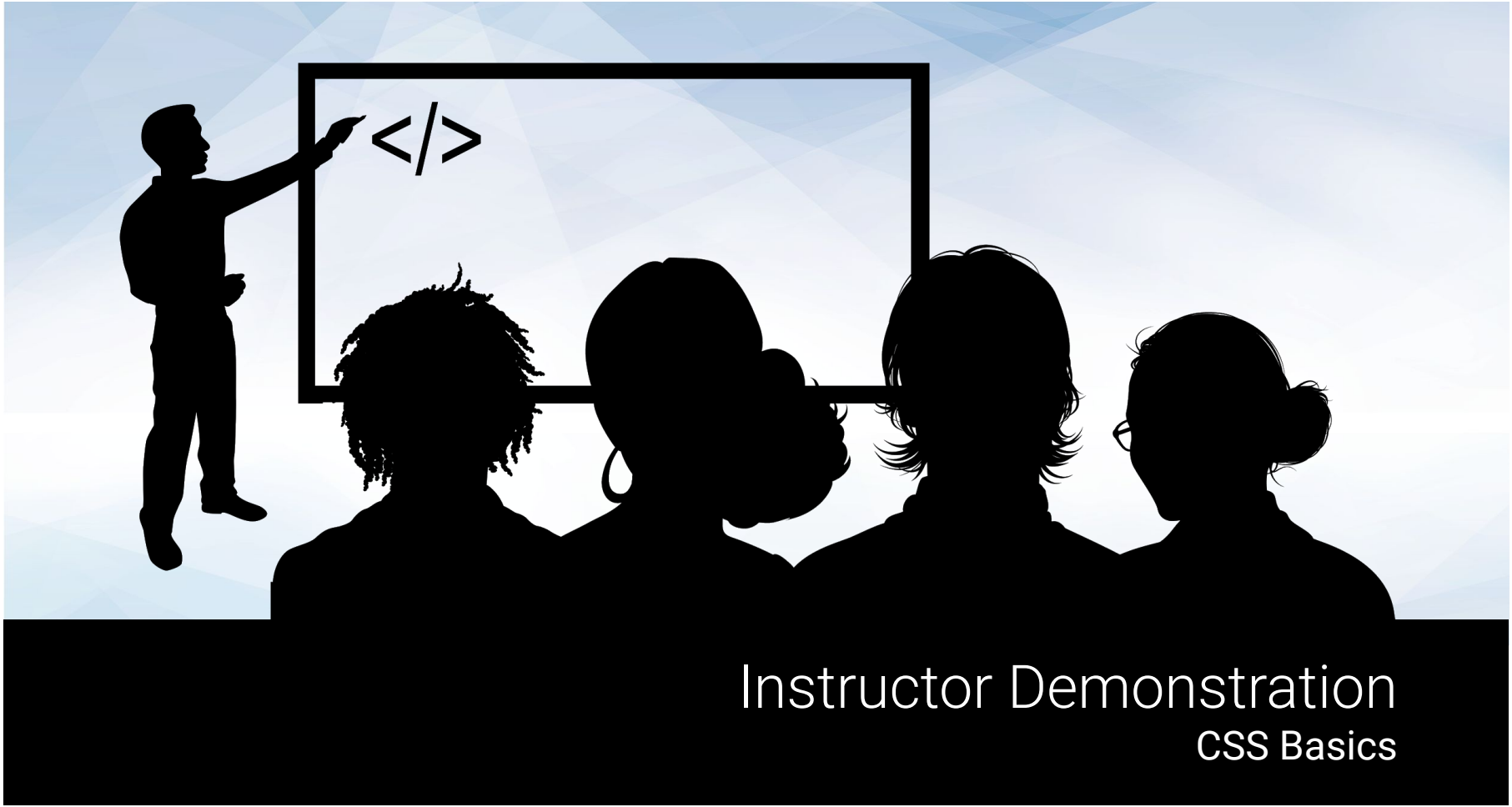
`<p class="bigBlue">Header</p>`	`.bigBlue`
	`{`
	`font-size: 100px;`
	`color: blue;`
	`}`

Key CSS Attributes

Font and Color:	
color:	sets color of text
font-size:	sets size of the font
font-style:	sets italics
font-weight:	sets bold
Alignment and 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

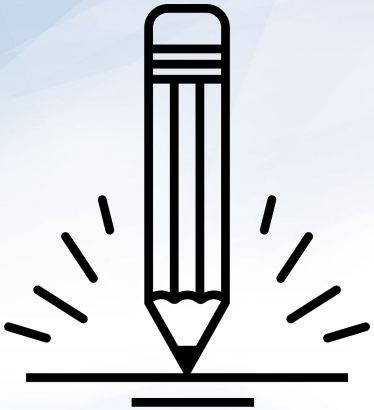
Powerful Duo

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



Instructor Demonstration

CSS Basics



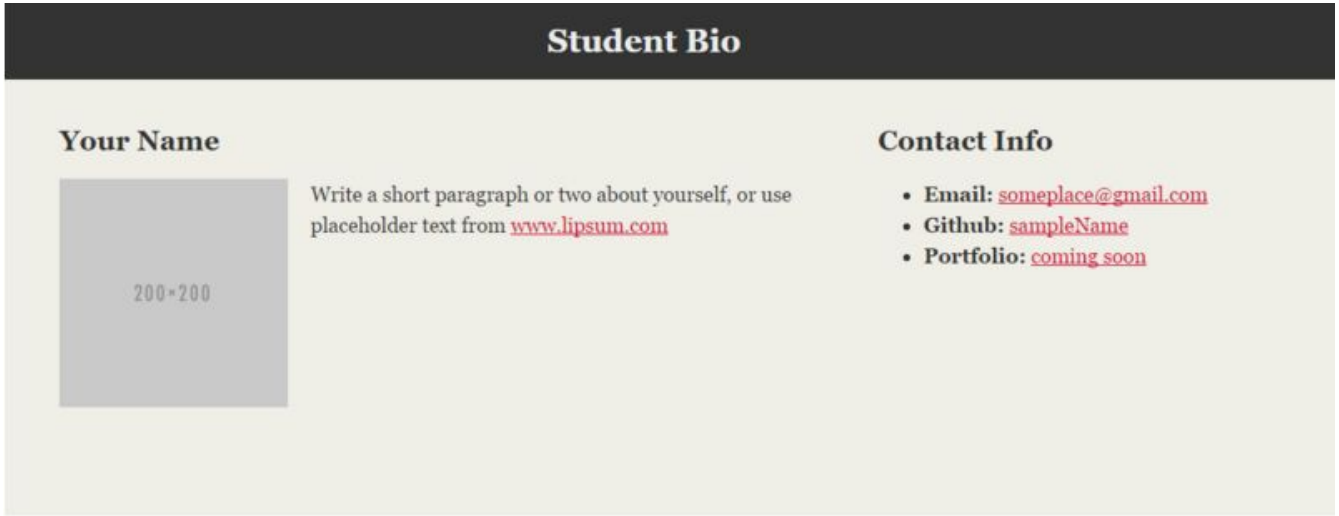
Activity: CSS-Styled Bio Page

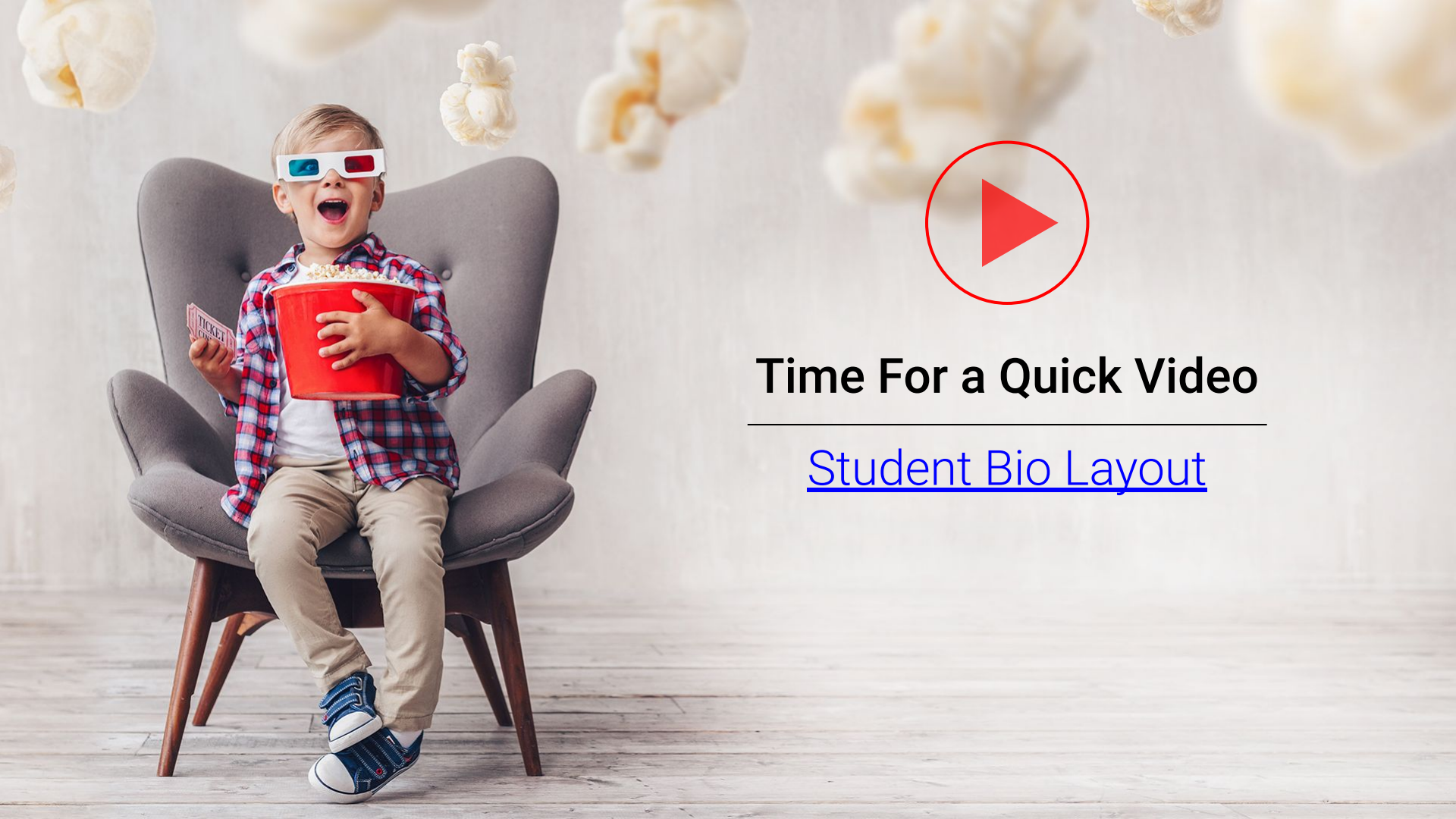
In this activity, you'll upgrade your previous HTML bio page using CSS style rules. Once you're done, commit and push your changes to GitHub.

(Additional instructions will be sent via Slack)

Suggested Time:
20 minutes

Activity: CSS-Styled Bio Page





Time For a Quick Video

Student Bio Layout

Still a Bit Confused?

- Remember the video guides for key activities (such as the last one).
- If you EVER feel like you are falling behind, use the video walk-throughs to catch 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 however we can.

Questions?