Start Slide

[Presentation begins after this one!]

Welcome to

Learn to Code

Last updated: 2/3/2017

powered by galvanize

About Galvanize

Dynamic learning community for technology

- Web Development
- Workspace
- Data Science
- Networking

To learn more, visit galvanize.com

powered by galvanize

For more information

Email **Lee Ngo** at
lee.ngo@galvanize.com
or
Visit our website at
galvanize.com

powered by galvanize

But first...

powered by galvanize

Let’s get to know each other

Turn to the person next to you and ask:

1) What is your name?
2) Why did you come here?
3) What’s the nerdiest thing about you?

You have 2 minutes to complete this mission.

powered by galvanize

Workshop #1

Intro to HTML & CSS

powered by galvanize

About this Workshop's Architects

Graham McBain

@grahammcbain

Graduate of the Full Stack Immersive Program (g3)

About this Workshop's Architects

Lee Ngo

github.com/lee-ngo

Galvanize Evangelist based in Seattle

Made a Game of Thrones text adventure game

About this Workshop's Architects

Brian Lee

g40 Student - Seattle

Once aspired to be a wartime photojournalist

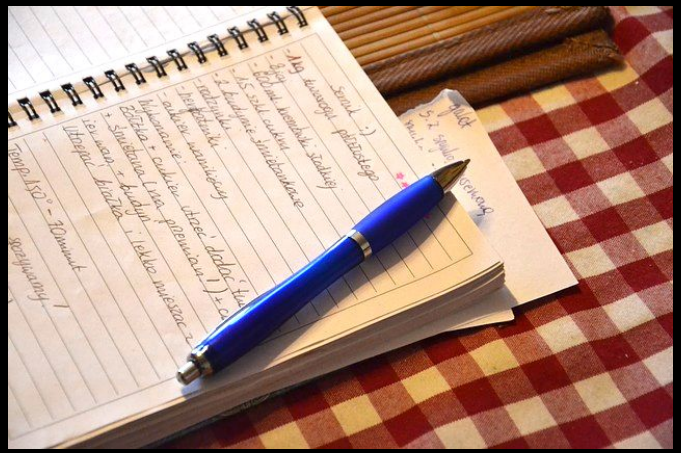
In this course you will learn

- **Set up** your computer for web development
- Overview of basic **HTML** concepts
- Overview of **CSS** concepts
- Working in the **sandbox**

Gut check, Galvanize style!

- This course is for beginners
- Feel free to move ahead
- Help others when you can
- Be patient and nice
- We'll all get through it!

What web coding is (really!)



&

Recipes to give to your computer to “cook” up some awesome things for you online

Setting up your computer

(Brace yourself…)

We’re going to set up...

- A web browser
  - To see our work as we’re editing it
  - Recommended: Google Chrome
- A text editor
  - To edit your code on your computer
- Our tutorial files
  - To have handy in case you want to work ahead

Install a web browser!

We highly recommend Google Chrome for its developer-driven features, esp. “Inspect Element”

google.com/chrome

Install a text editor!

Download a text editor to help you write code:

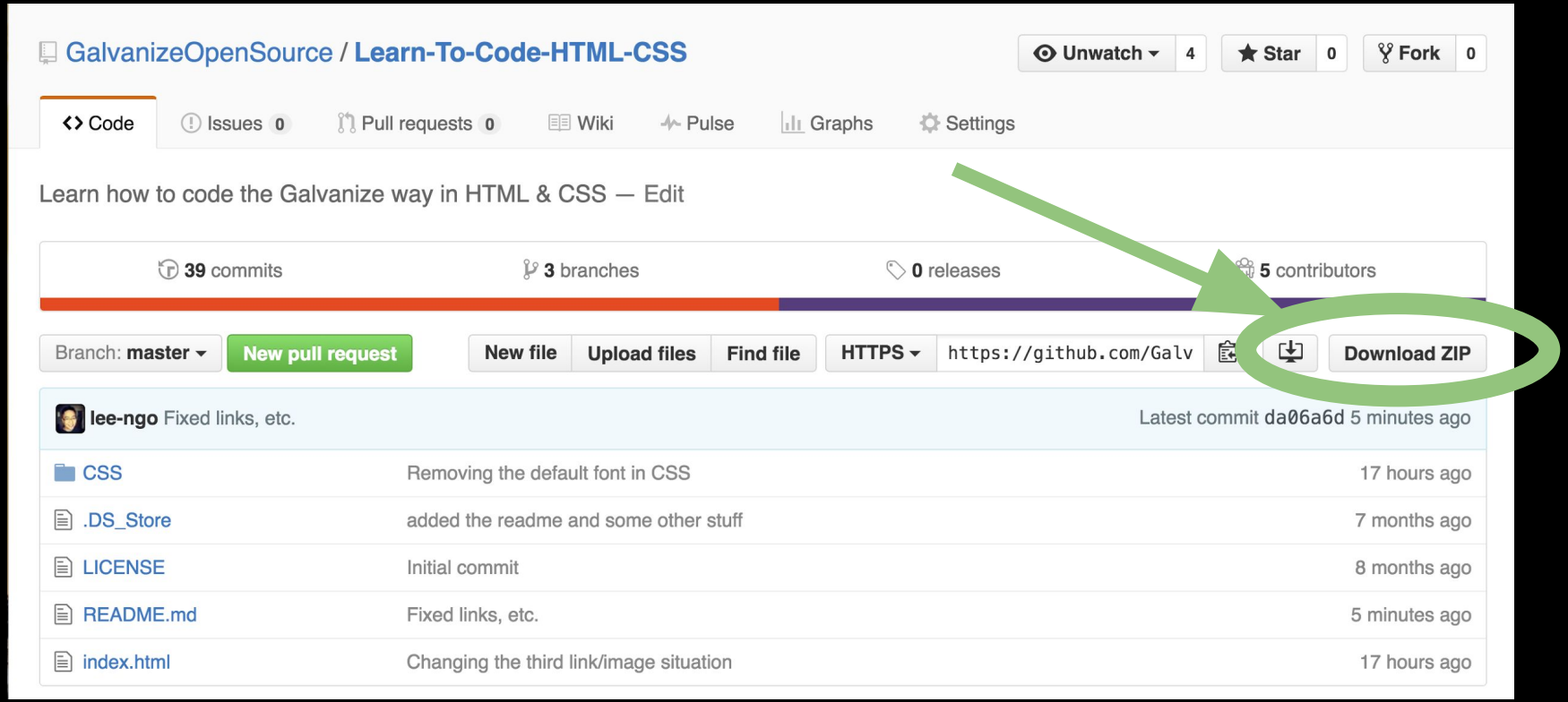
We recommend

Atom: atom.io

Download the code!

1. Go to: github.com/brianjleeofcl/Learn-To-Code-HTML-CSS-v2/
2. Download the zip file of our code and unzip
3. Open the **whole folder** in your text editor
    a. index.html
    b. CSS/style.css
4. Open the index.html file in your browser

The Download ZIP is right there...



Alternative: Use CodePen

1. Go to codepen.io
2. Go to this link: http://codepen.io/lepngo/pen/kkkzpd
3. Focus on the HTML and CSS windows
4. Copy the code

Pictures of Puppies

Setting up your computer can be tricky...



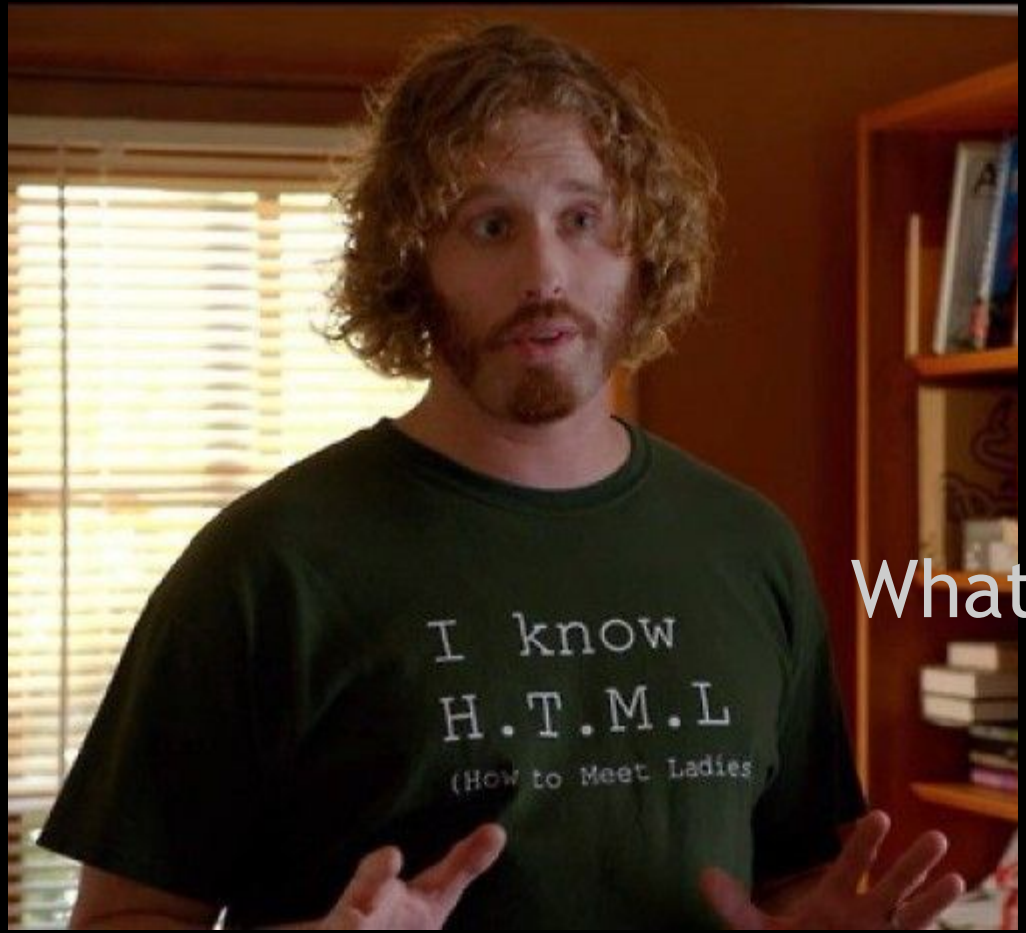
If you've done the following:

- ❑ Install Google Chrome
- ❑ Install a text editor
- ❑ Downloaded and opened the tutorial files

You're ready to move on to the next step!

In this course you will learn

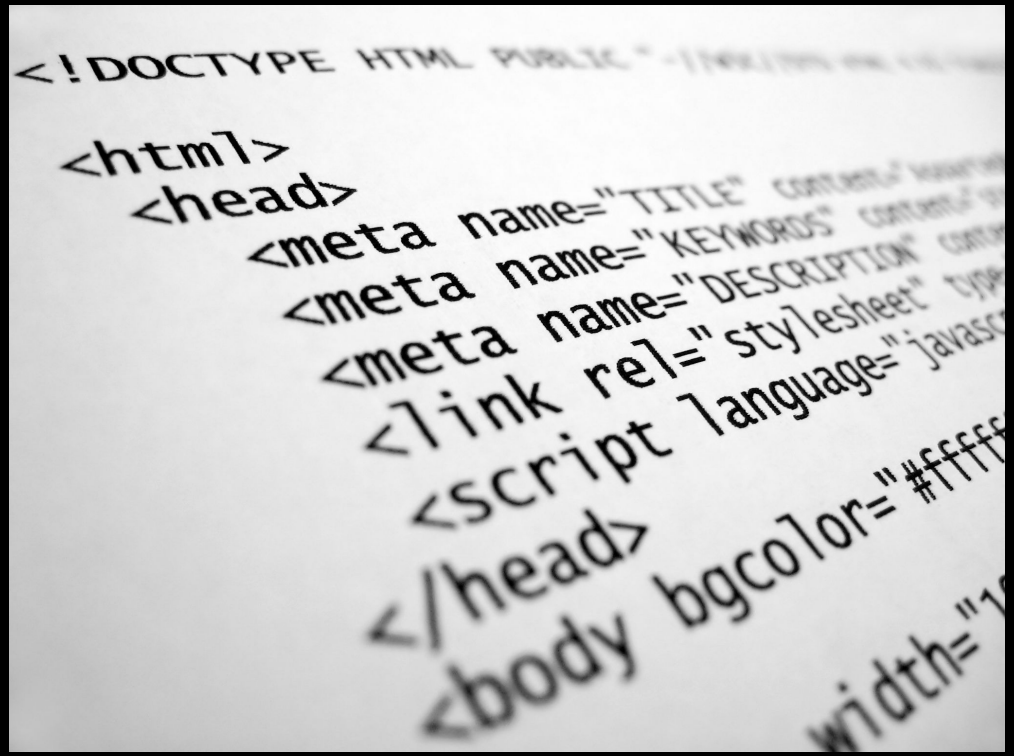
- ~~**Set up** your computer for web development~~
- Overview of basic **HTML** concepts
- Overview of **CSS** concepts
- Working in the **sandbox**



H.T.M.L.

What does it stand for?
(No, not that.)

Hyper Text Markup Language



"the building blocks of the internet"

3 Basics of HTML syntax

<Tags>

Attributes=””

Element

```
Element = <tag attribute=”blahblah”>
content</tag>
```

<Tags> - when HTML is “activated”

Almost everything in HTML needs to start and end with a tag

Wrapped like layers of an onion

<Opened> and </closed>

e.g. ”Hello!”

* Not every tag is like this!

Tags used mainly for structure

<html> designates document as HTML

<head> contains undisplayed information

<body> contains displayed information

<header>,<main>,<footer> denotes where on page elements belong

 creates a division of elements, often for content-neutral grouping

<nav> contains navigation menus

<section> contains a section

Tags used mainly for content:

- <h1> - <h6> create section headings
- <p> creates paragraphs
- (anchor), activates a link in the page
- , <ol> creates lists
    - contains items in lists

Self-closing Tags

<img src=""> creates an image in the page

 creates a break in the content

<input type=""> creates an input field

Attributes=”” - do more with tags

```
<p class="foo">This is a paragraph.</p>
```

With attributes, we can inform the browser on what to do with a tagged piece of content.

Common Attributes=””

href=”” - hyperlink reference to an internal or external link

src=”” - source file to an image, video, etc.

style=”” - add some color, font, margins, etc.

** ^ There’s a MUCH better way to do this via CSS - more on that later!

LET’S CODE!

Do the following:

- Name of your website in the header (hint: consider using a heading tag)
- A picture in the figure section of your main page
- Two very short paragraphs in the middle section
- An ordered list and an unordered list in the last section
- A short copyright notice in the footer

LET'S CODE!

Feeling ambitious?

Add the following:

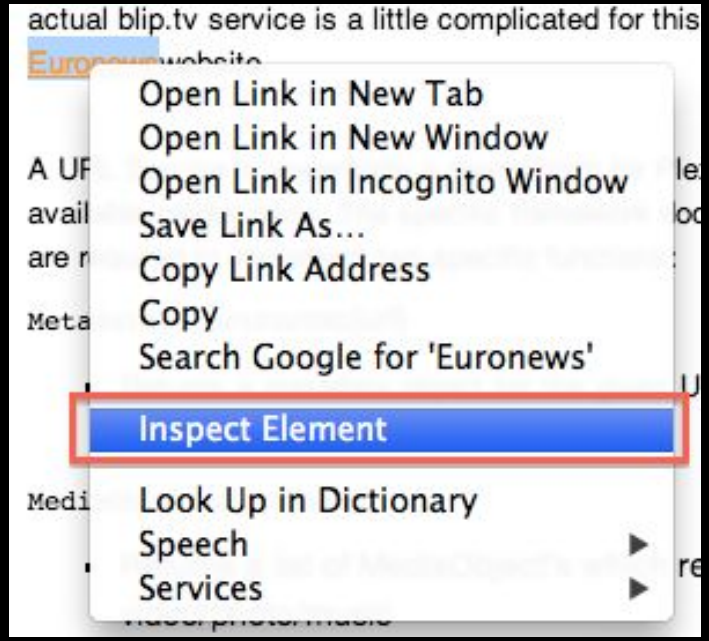
- Insert a caption under your picture
- Insert a youtube video in your main page
- Add a link to your email address in the copyright notice
- Add a navigation bar in the header of the page

Elements: Tags + Attributes + Content

Put it all together!

Elements are what HTML is all about.

When we look into the HTML of any page (in Chrome), we "inspect element"



If you've done the following:

- ❑ Attempted some basic HTML editing
- ❑ Bonus: attempted advanced HTML editing
- ❑ Inspected your elements in Chrome

You're ready to move on to the next step!

But how do we make HTML… better?

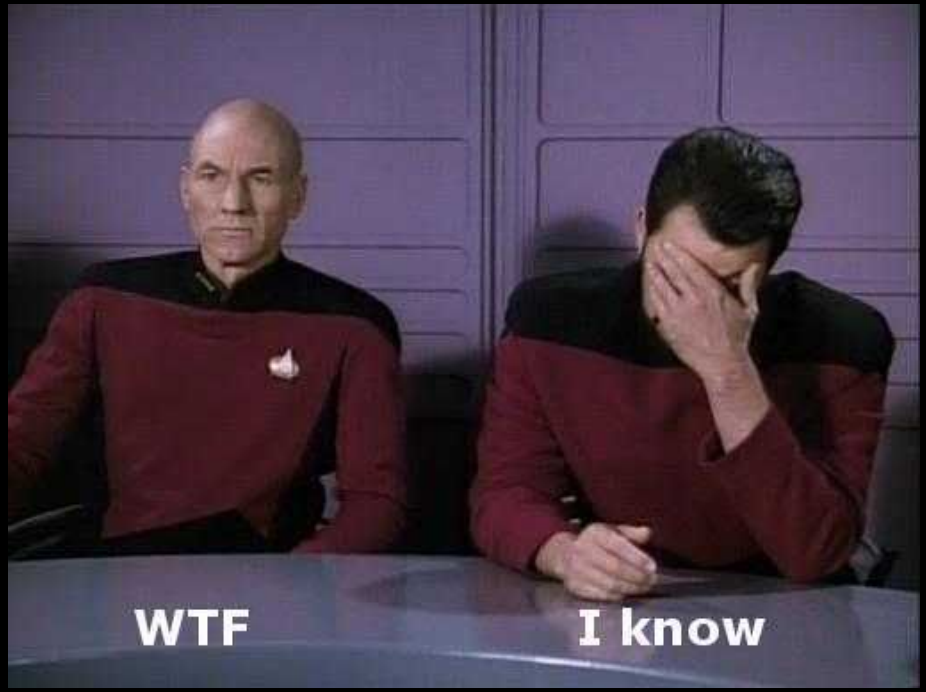
(This is what pure HTML looks like …ew.)

In this course you will learn

- ~~**Set up** your computer for web development~~
- ~~Overview of basic **HTML** concepts~~
- Overview of **CSS** concepts
- Working in the **sandbox**

C.S.S

What does it stand for? (Please don’t Google it.)



Cascading Style Sheets

(what...?)

Cascading - prioritizing certain values over others

Style - focusing on layout, colors, fonts, etc.

Sheets - another name for the file we use here

Before CSS, the internet was UGLY

The first White House website in 1994, when Al Gore “invented the internet”

1998: let’s separate presentation stuff into different docs!

What does a CSS file look like?

A “cascade” of objects with attributed styles

Elements: e.g. h1, div, body, a - default HTML

IDs: everything that starts with a “#”

Classes: everything that starts with a “.”

```
8  body {
9      color: #414141;
10     background: url(../images/bg.jpg) repeat-x #ebe8df;
11     font-family: Arial, Helvetica, sans-serif;
12     line-height: 120%;
13     font-size: 12px;
14 }
15
16 a:link, a:visited {
17     color: #685966;
18     text-decoration: underline;
19 }
20 a:hover {
21     color: #2b212c;
22 }
23 .article_separator {
24     line-height: 5px;
25     height: 5px;
26     font-size: 5px;
27 }
28 /* SITE WIDTH
29  *-----------------------------*/
30 .rht_container {
31     width: 1020px;
32     margin: 0 auto;
33     margin-top: 25px;
```

Syntax of CSS

```
h1 {  // this is either an element, class, or ID
    font-size: 24px; // syntax is name: value;
    font-weight: bold;
    color: #000000; // hexadecimal, RGB, etc.
}
```

Space doesn’t matter, but “onion” rules apply

id vs class

id - attributes something to one thing ONLY

noted with a “#” symbol in CSS

- HTML: Lee Ngo
- CSS: #leesName { color: white; }

When and why do we use this?

id vs class

ids are used to direct functions to unique elements in the HTML so that there's no confusion

e.g.: clicking to a specific part of page

id vs class

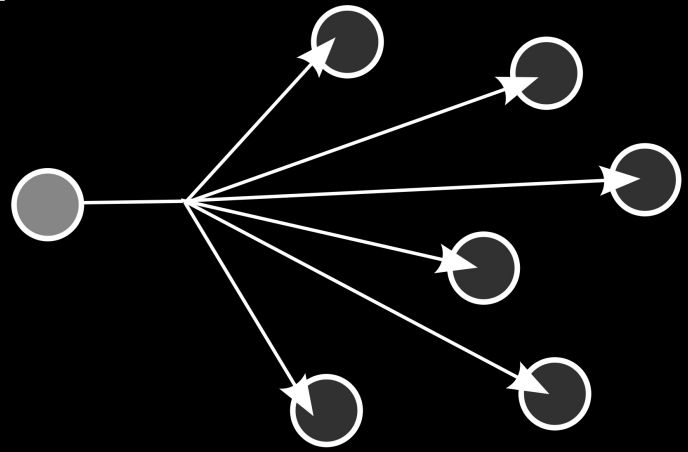
class - attributes something to multiple things noted with a “.” symbol in CSS

- HTML: Lee Ngo
- CSS: .ninja { color: black; margin: 10px; }

When and why do we use this?

id vs class - very important!

classes are used to change or affect multiple items in an HTML document at once



e.g. everything with class=”ninja” should have the same attributes

HTML & CSS harmony - do both!

HTML

1) Include your CSS in a link in the <head></head>
2) Add attributes in your tags e.g. id=”” or class=””

CSS

1) list out your elements, ids, and classes here
2) modify attributes only in your CSS

LET'S CODE!

1. Using height and width, resize the image on your page
2. Make your header look distinct from the rest of the site. Add a border? Different background colors? Some margin? Plenty of ways to accomplish this; try different things and decide what you like.
3. Try using text-align with left, center and right

LET'S CODE!

Feeling ambitious?

1. Can you think of a way to have the two paragraphs in the middle section appear side-by-side? (hint: display)
2. Can you think of a way to apply the same style to even-numbered elements on the ordered list?

If you've done the following:

- ❑ Complete the basic CSS exercises
- ❑ Bonus: do the more advanced exercises

You're ready to move on to the next step!

In this course you will learn

- ~~**Set up** your computer for web development~~
- ~~Overview of basic **HTML** concepts~~
- ~~Overview of **CSS** concepts~~
- Working in the **sandbox**

Play around in the sandbox! Try to...

- Change the name of the site to...whatever!
- Change all the navigation links & section headers
- Replacing my images with your own images

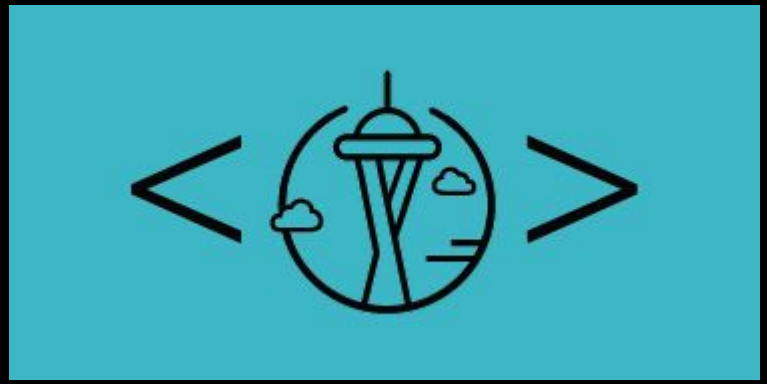
In this course you will learn

- ~~**Set up** your computer for web development~~
- ~~Overview of basic **HTML** concepts~~
- ~~Overview of **CSS** concepts~~
- ~~Working in the **sandbox**~~

You did it!

You are now a coder.
Welcome to the cool kids club.

Keep the party going!



Come back for more!

Sign up via the

Learn to Code Meetup

meetup.com/

learn-code-seattle

Try some more stuff at home!

Check out our online curriculum at
workshops.galvanize.com

Web Development Workshops Available

Web Development
Foundations in JavaScript
an 8-week part-time course

Classes begin on Feb 13!
Use code ‘SEATTLE30’ for 30% off!
Email amara.wilson@galvanize.com for more information.

powered by galvanize

About Web Development Immersive

- 24 Week Full-Time Program
- 91% Job Placement Rate within six months
- Average starting salary: $77,000 per annum
- Scholarships available for those who qualify

Next cycle begins in February 2017 - **apply now!**

powered by galvanize

Thank you for coming to galvanize

Email **Lee Ngo** at
lee.ngo@galvanize.com
or
Visit our website at
galvanize.com

This course has been brought to you by the evangelists of Galvanize.

Workshop #1

Intro to HTML & CSS

powered by galvanize

End Slide

[Appendix begins after this one!]

Learn to Code Contributor

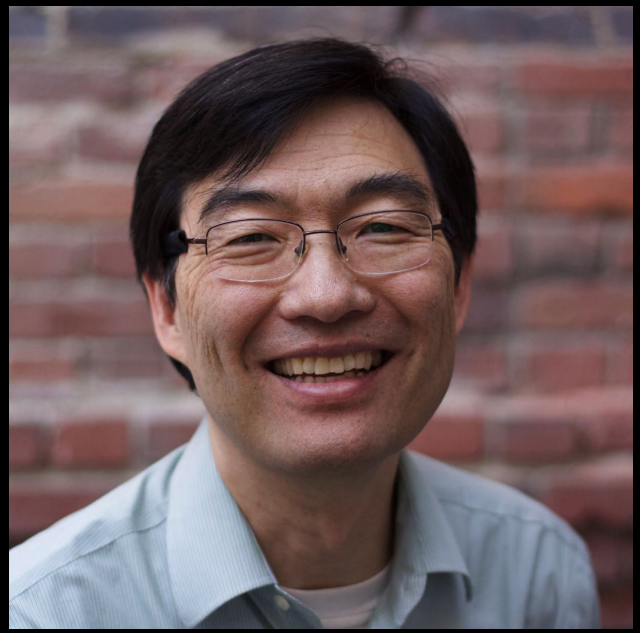
Thacher (@thachert1d)

github.com/thachert1d

Full Stack Student (g21) at Galvanize Seattle

Wants to solve diabetes with code

Learn to Code Contributor



Michael Park

that.michael.park@gmail.com

G28 student - Seattle

Michael started programming back when punched cards were a thing.

Learn to Code Contributor

Jake Bell

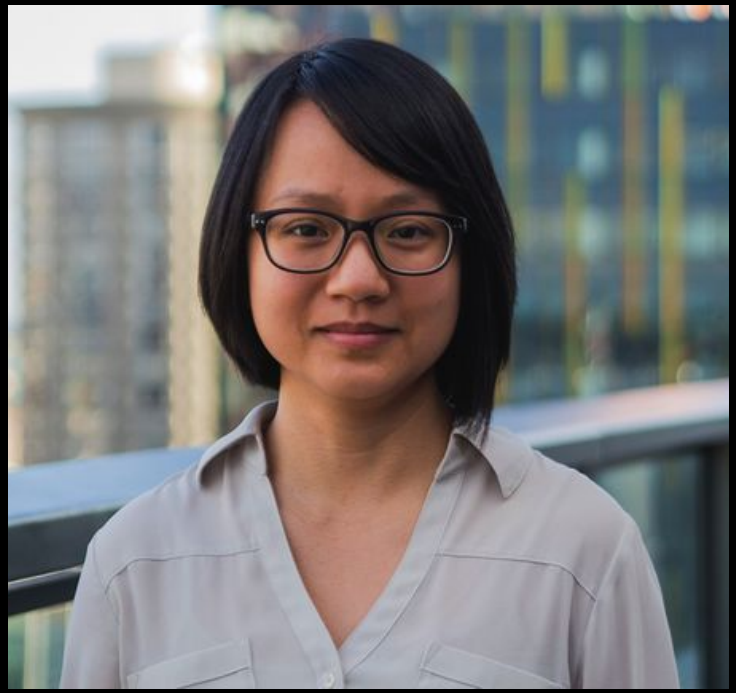
github.com/jakebrbell

G28 Student - Seattle

Once interned at

The White House

Learn to Code Contributor



Hien Dang

github.com/hienpd

G28 student - Seattle

Once a non-profit marketing professional

Hien knows kung-fu (for realsies!)

Learn to Code Contributor

John Carney

github.com/jcquery

g28 Student - Seattle

Has eaten live octopus more than once; does not recommend.

Learn to Code Contributor

Ross Todd

github.com/rsstdd

g34 Student - Seattle

Spent the last year exploring the mountains of Colorado

Learn to Code Contributor

Rachelle Rathbone

github.com/rrathbone

g34 student - Seattle

Hates having her photo taken, can’t live without peanut butter or bacon.

About this Workshop's Instructor

Bryan Brophy

github.com/brybrophy

g28 Student - Seattle

He has showered in the locker rooms of the Denver Broncos and the San Antonio Spurs

Learn to Code Contributor

Hiromi Heider

g40 Student - Seattle

Enjoys coding while eating nerds rope and petting cute pit bulls

Learn to Code Contributor

Minh Chau

g40 Student - Seattle

Resident code obfuscation specialist in his cohort

Learn to Code Contributor

Alicia Gyori

g40 Student - Seattle

Previously a web designer addicted to writing stories