
Network ID: GA-Guest

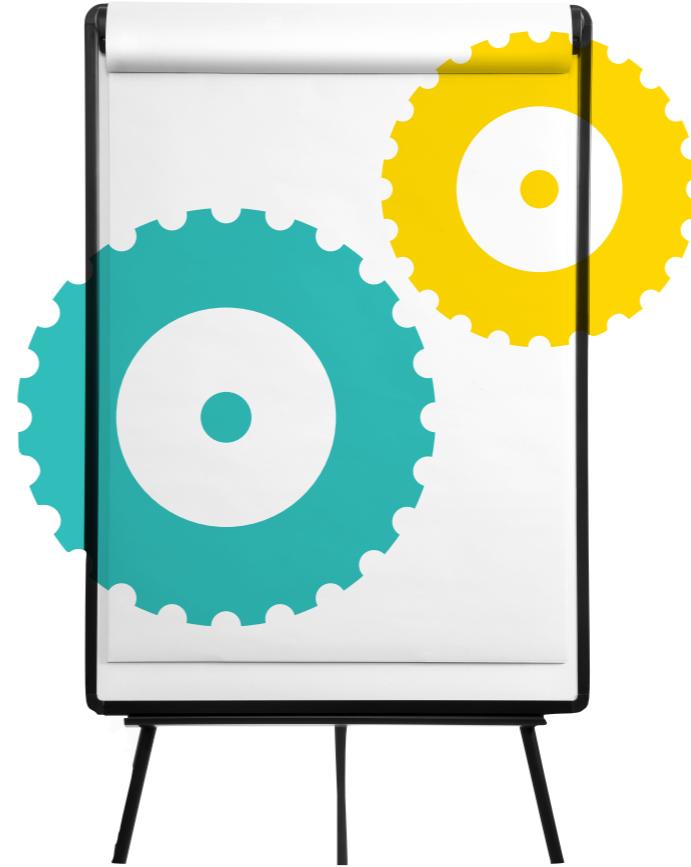


Password: yellowpencil

JAVASCRIPT 101

ORIENTATION AGENDA

- Your Producer
- General Course Info
- GA Mission
- Student Experience



WELCOME!

MEET YOUR COURSE PRODUCER

EVAN PONCHICK
EDUCATION PROGRAMS PRODUCER
evan@galvanize.com



JavaScript 10151

- November 1st - January 19th
- Tuesdays and Thursdays 6:30-9:30pm

ABOUT GENERAL ASSEMBLY



FRONT-END WEB DEVELOPMENT ORIENTATION

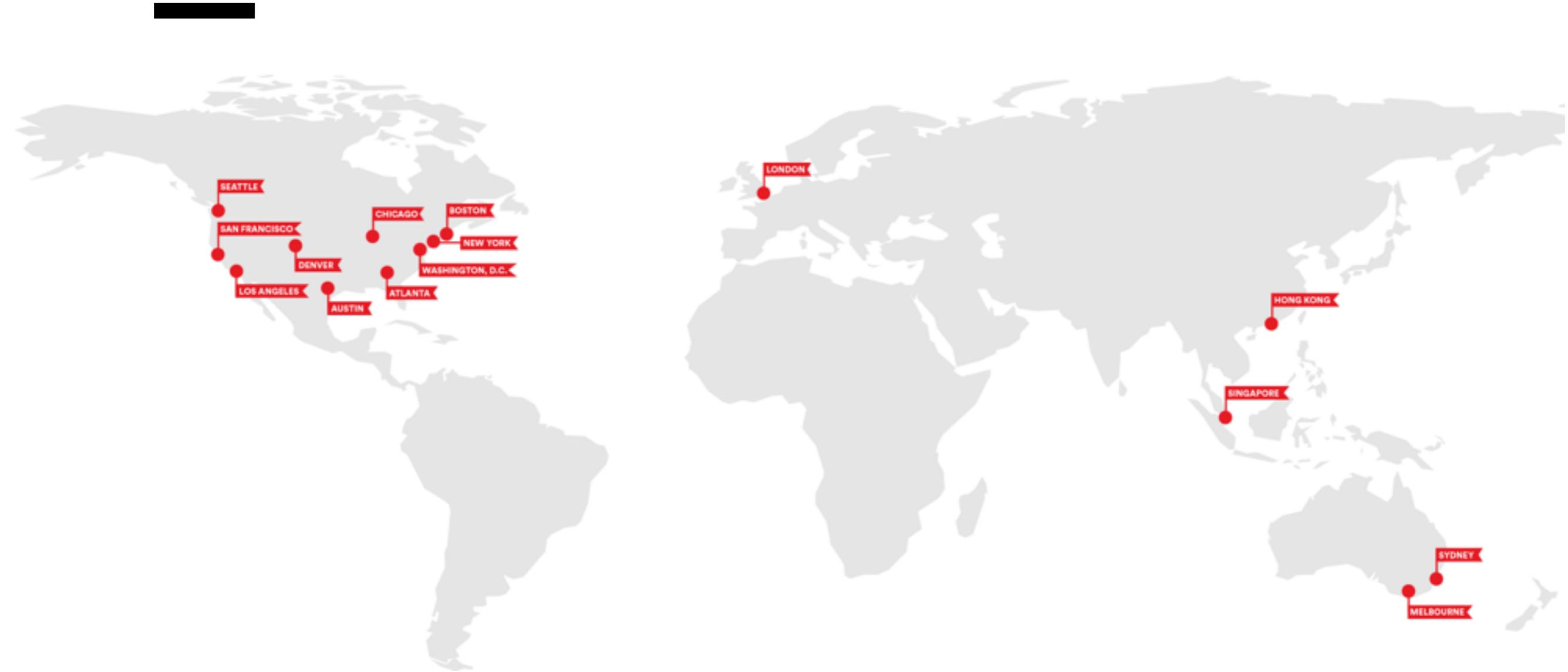


—

**GENERAL ASSEMBLY IS A GLOBAL
COMMUNITY OF INDIVIDUALS
EMPOWERED TO PURSUE THE
WORK WE LOVE.**

—

GENERAL ASSEMBLY'S MISSION IS
TO BUILD OUR COMMUNITY BY
TRANSFORMING MILLIONS OF
THINKERS INTO CREATORS.





Front End Web Development

All Cities

trendy

TABBOULEH NYC

by Jack Breslauer

FEWD

Svadhishtana | Sacral
Muladhara | Root

Sahasrara | Crown

/thought/

creativity, beauty, wisdom, generosity,
knowledge, spiritual connection

Sahasrara or crown chakra is generally
considered to be the state of pure
consciousness within which there is

CHAKRA COLOR SF

by Rita Troyer

OCEAN FACTS DC

by Colleen O'Reilly

FEWD

gogobones

Roll About

Roll

Specify the number of die and select the die type. Include a second die set or modifier to create advanced rolls.

Setup

Die Sets: 1 2

Die Set 1:

3 d6

Modifier:

+ 3

Results

Result (3d6 + 3)

10

Range: 6 to 21

Median: 13.5

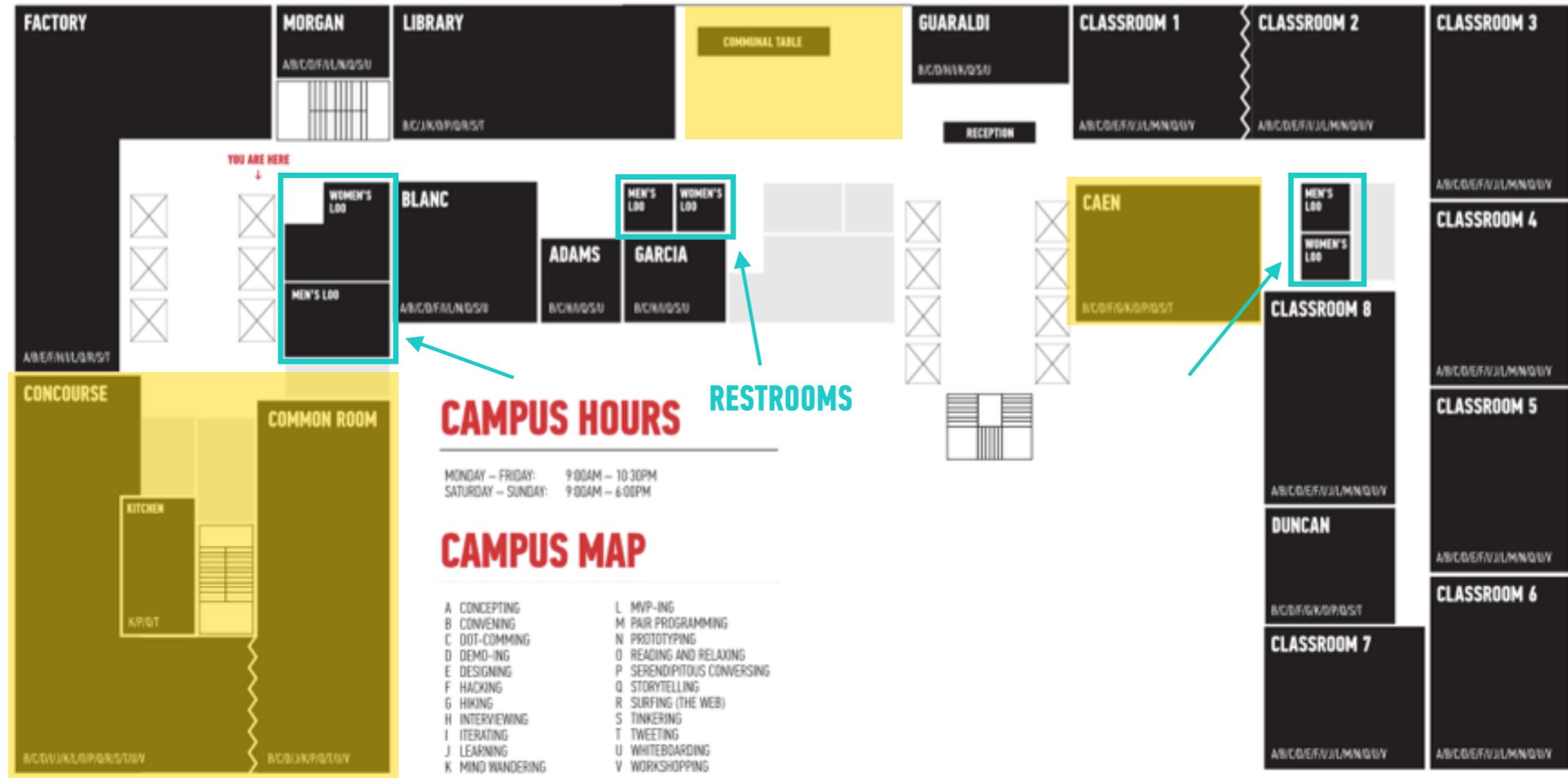
GOGOBONES CHI

by Christopher Zalek

STUDENT EXPERIENCE

FRONT-END WEB DEVELOPMENT ORIENTATION





STUDENT SPACE

GA STUDENT EXPERIENCE

COMMUNITY

- Kitchen
- Coffee
- Snacks
- Student Work Spaces



GA STUDENT EXPERIENCE

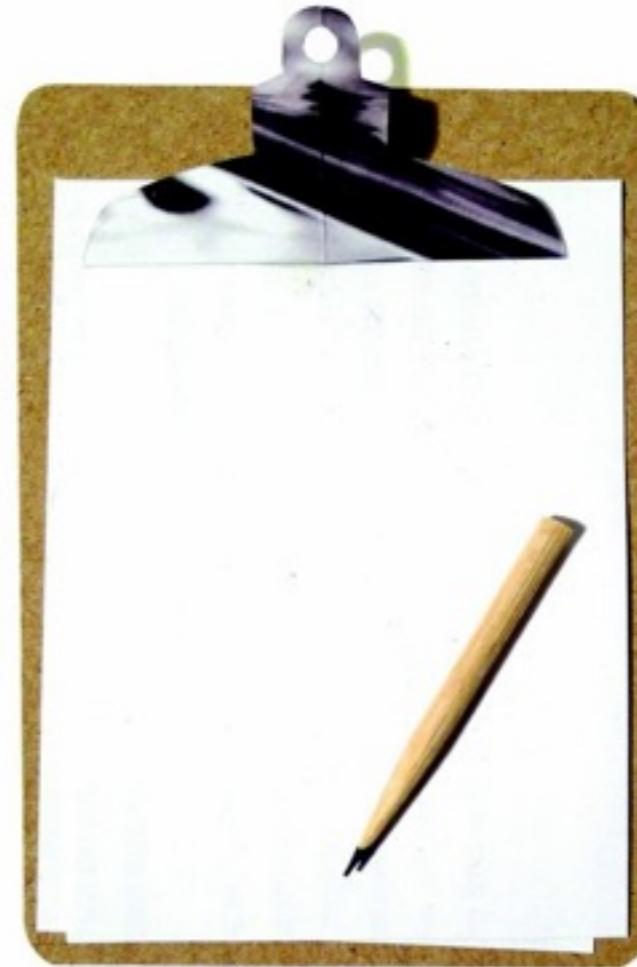
CLASS/WORKSHOP VOUCHERS

- › 15% off all classes and workshops
code: currentstudentdiscount15

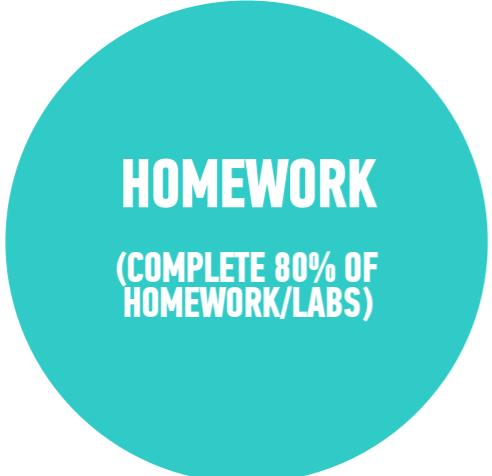


IN CLASS + FEEDBACK

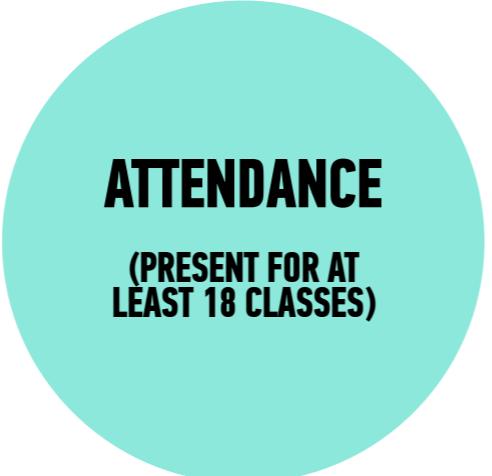
- Ask questions
- Share with peers
- Complete exit tickets after each lesson
- Complete mid & end of course surveys
- Make friends :)



SUCCESSFUL GRADUATE



HOMEWORK
(COMPLETE 80% OF
HOMEWORK/LABS)



ATTENDANCE
(PRESENT FOR AT
LEAST 18 CLASSES)

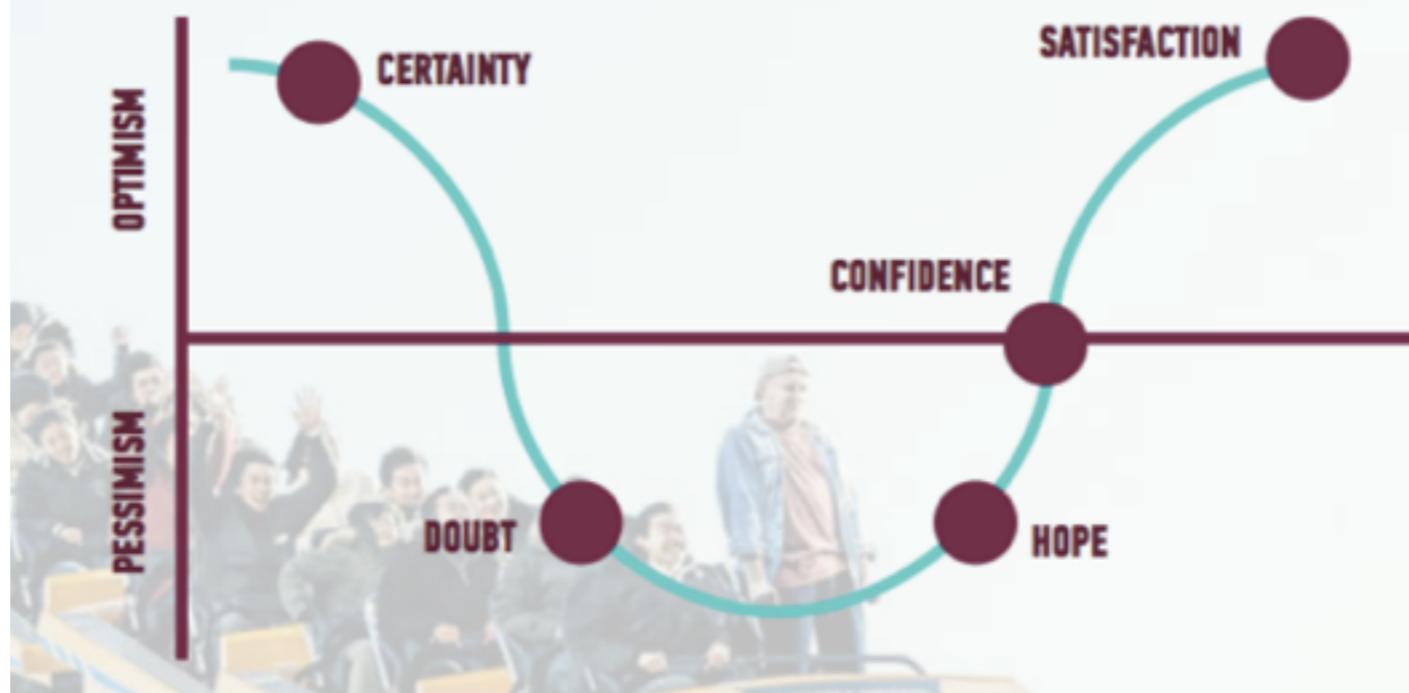


**FINAL
PROJECT**



**COMMUNITY
ENGAGEMENT**

THE LEARNING ROLLERCOASTER



evan@google.com

evan@google.com

Hello!

HELLO!



SARAH HOLDEN — INSTRUCTOR

[LinkedIn](#)

[Instagram](#)

[Facebook](#)

INTRODUCTIONS

1. Name
2. What do you do?
3. What brought you here tonight?

GREETINGS //
GA EXPERIENCE

HTML BASICS

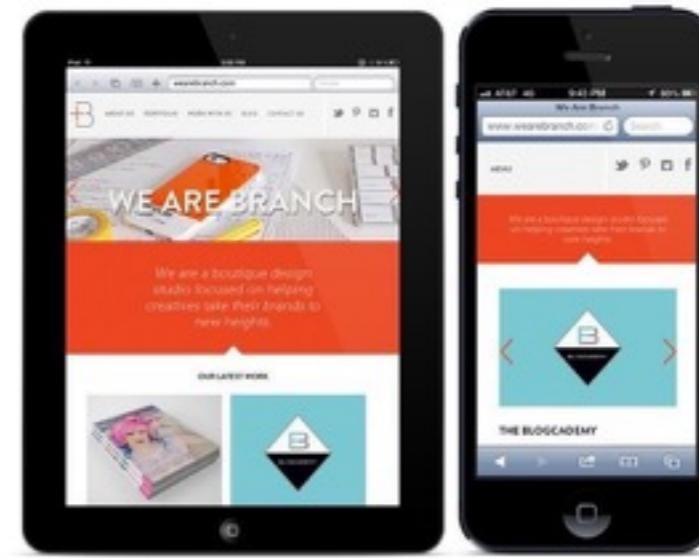
LEARNING OBJECTIVES

- › Gain an overview of the JavaScript landscape and its placement in the web ecosystem.
- › Practice programmatic thinking by writing pseudo-code.
- › Write expressions that both assign and evaluate variables.
- › Explain the difference between jQuery and vanilla JS.
- › Register and trigger event handlers for jQuery events.

FRONT-END OVERVIEW

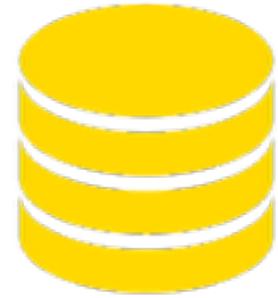
WEB PRODUCTION WORKFLOW — FRONT END

- ▶ Take a design and turn it into code and assets
- ▶ Uses HTML/CSS to create the structure for a page and add styles
- ▶ Add interactions with JavaScript
- ▶ Other responsibilities: accessibility, performance, cross-browser and cross-device functionality

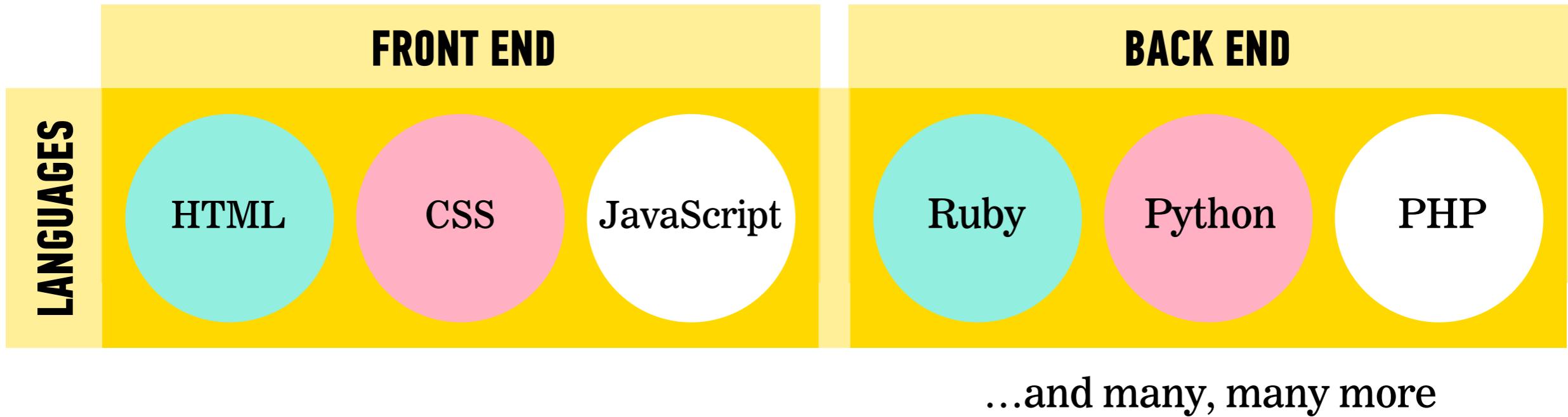


WEB PRODUCTION WORKFLOW — BACK END

- ▶ Give the application/webpage a 'memory'
- ▶ Handle page request and send back the right page
- ▶ Talking to the database - getting data, adding and updating data
- ▶ Authenticate users



FRONT END VS. BACK END



WHAT DOES FRONT END EVEN MEAN?

“A mix of programming and layout that powers the visuals and interactions of the web.”

WEB PRODUCTION WORKFLOW — FRONT END

- The Front End is what the user sees.
- It powers the visuals and interactions of the web.
- It is meant to be pretty, but doesn't always happen that way.
- Made up of HTML, CSS and Javascript.

ACTIVITY



KEY OBJECTIVE

Describe the role and responsibilities of the front-end developer. How do these differ from those of the back-end developer?

TIMING

1 min

1. Turn to partner and discuss

2 min

2. Share with rest of class

BONUS

Come up with an analogy!

HTML / CSS / JS

FRONT-END

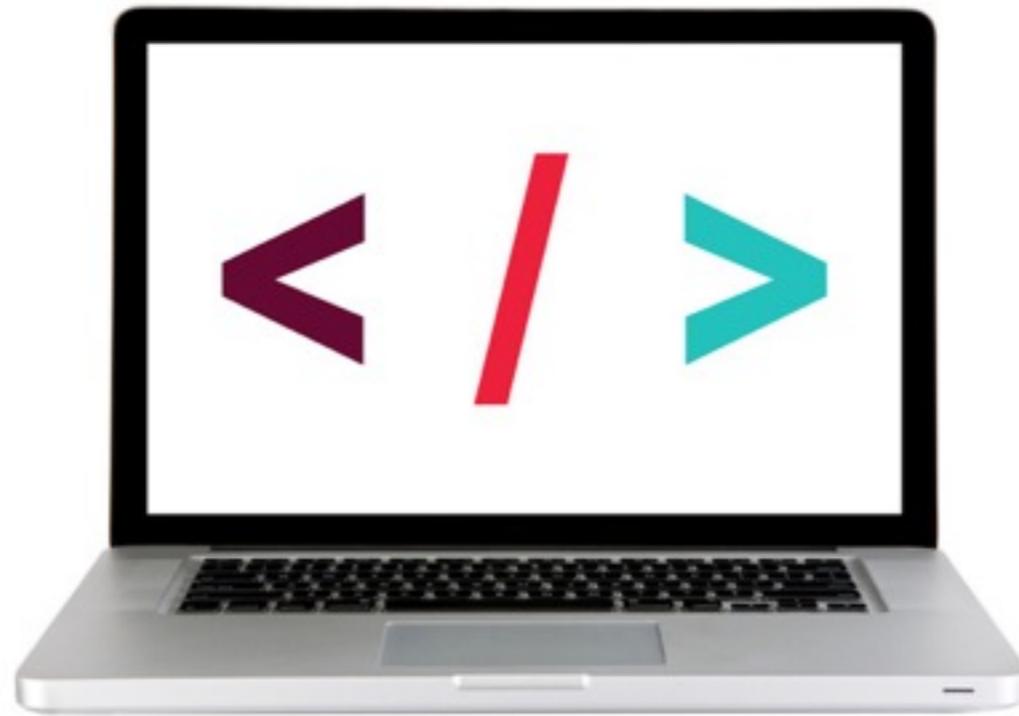
HTML

CSS

JavaScript



LET'S TAKE A CLOSER LOOK



WHAT IS JAVASCRIPT?

WHAT JAVASCRIPT CAN DO!

1

Access
Content

2

Modify
Content

3

Program
Rules

4

React to
Events

WHAT JAVASCRIPT CAN DO!

1

Access
Content

2

Modify
Content

3

Program
Rules

4

React to
Events

You can use JS to select any element, attribute or text from an HTML page.

For example:

- Find out what the user entered into a text input when they submit a form
- Find out whether the user checked a checkbox

WHAT JAVASCRIPT CAN DO!

1

Access Content

2

Modify Content

3

Program Rules

4

React to Events

You can use JS to add elements, attributes and text to the page (or remove them)

For example:

- Add an error message below a form
- Change the size, position, color, or other styles for an element

WHAT JAVASCRIPT CAN DO - MODIFYING CONTENT

Please Enter Your Details

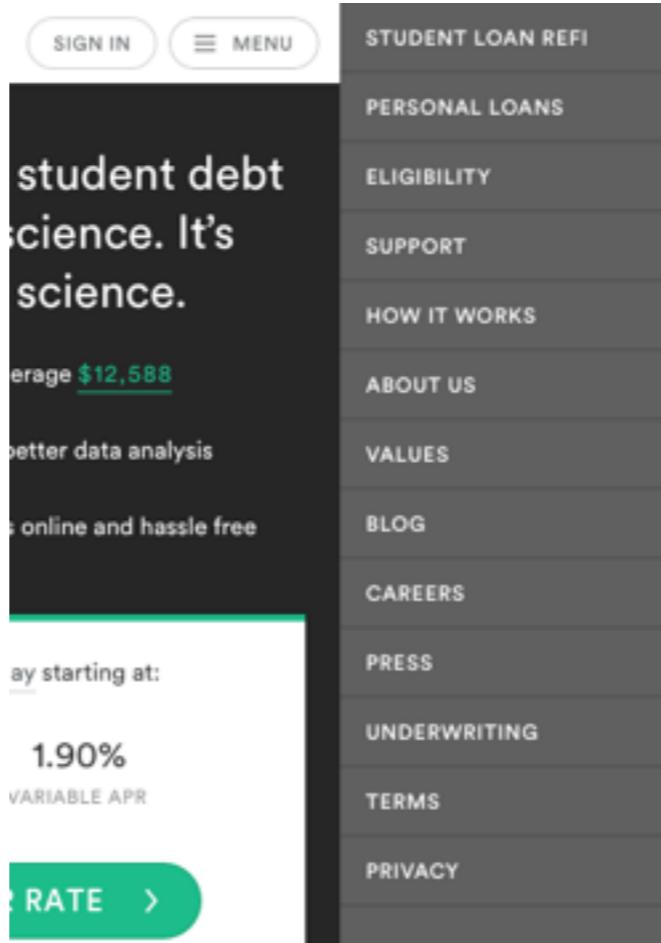
REQUIRED

Some fields below require your attention

FIRST NAME	Sarah
LAST NAME	Holden
HIGHEST DEGREE	Choose One
SCHOOL	Enter School Name
EMPLOYER	Enter Employer Name
JOB TITLE	Enter Job Title
ANNUAL INCOME	\$ Enter Your Annual Income (Not Household)
BANKING + INVESTMENT TOTAL	\$ Enter Estimated Total
STUDENT LOAN BALANCE	\$ Approximate Amount (\$5,000 Minimum)
STREET ADDRESS	Enter Street Address

Add an error message
(and styles) to a form

WHAT JAVASCRIPT CAN DO - MODIFYING CONTENT



Change the size, position, color, or other styles for an element

WHAT JAVASCRIPT CAN DO!

1

Access Content

2

Modify Content

3

Program Rules

4

React to Events

You can specify a set of steps (instructions) for the browser to follow.

For example:

- Have images/text fade in if the user has scrolled to a certain portion of the page
- Check to make sure the user has entered a valid email address into a form and display an error message if not

WHAT JAVASCRIPT CAN DO!

1

Access Content

2

Modify Content

3

Program Rules

4

React to Events

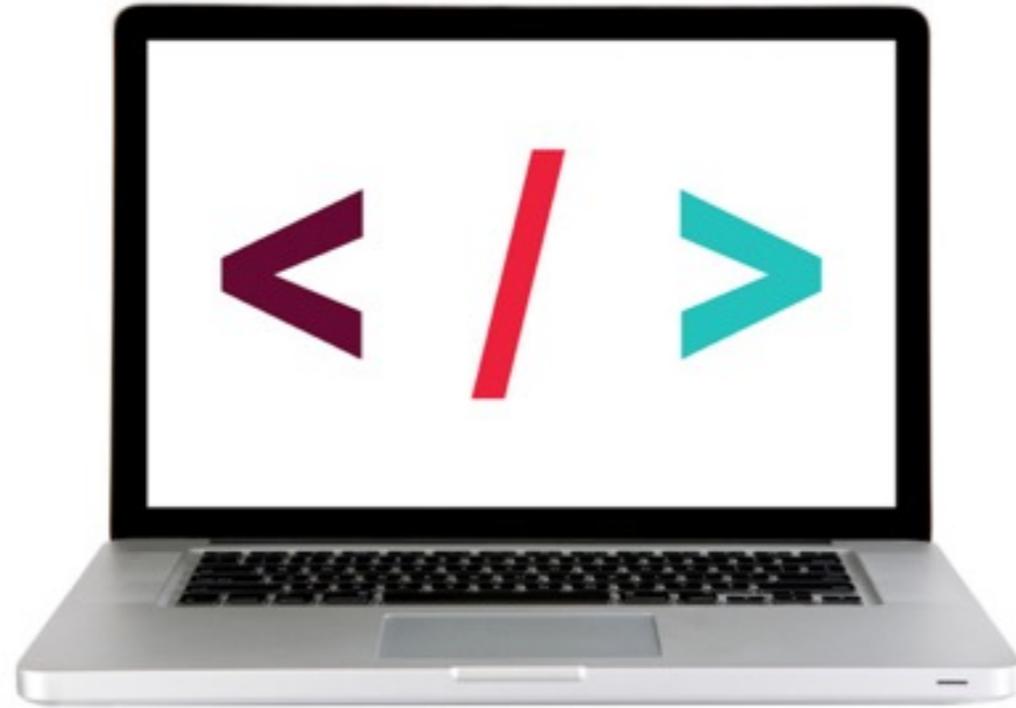
You can specify that a script should run when an event occurs

For example:

- ▶ When a button is clicked
- ▶ When the cursor hovers over an element
- ▶ When the user types information into a form

GET YOUR RATE >

LET'S TAKE A LOOK



<https://kinhr.com/>

ACTIVITY



TIMING

10 min

1. Let's visit this [codepen](#) together
2. Turn to someone next to you and as a team try to figure out on a high level what is happening
3. With your partner try to make it so that the slow button changes the bulb to yellow
4. With your partner try to make it so that the go button works
5. This exercise is simply to help introduce you to what JavaScript looks like, do not get caught up in all the details just yet

BONUS

Discuss: What does clearLights do? Is order important here?

INTRO TO PSEUDO CODE

PSEUDO CODE

- ▶ A way to 'plan out' your program before coding it
- ▶ A *detailed yet readable description* of what a computer program must do
- ▶ Expressed in plain english rather than in a programming language

THE IMPORTANCE OF PLANNING



PSEUDO CODE — SCORES

Take, for example, a program that lets a player know whether he or she has passed the current level. Pseudo code for this program might look like the following:

```
passingScore = 50 points
get playerScore

if playerScore >= passingScore
    display message "Current level: Passed"
otherwise
    display message "Current level: Failed"
```

PSEUDO CODE LAB - THERMOSTAT



DIRECTIONS

- ▶ With a partner, write pseudo code for an application that would monitor the room temperature and adjust it so the room remains at a certain temperature.

TYPE OF EXERCISE

- ▶ Groups of 2

TIMING

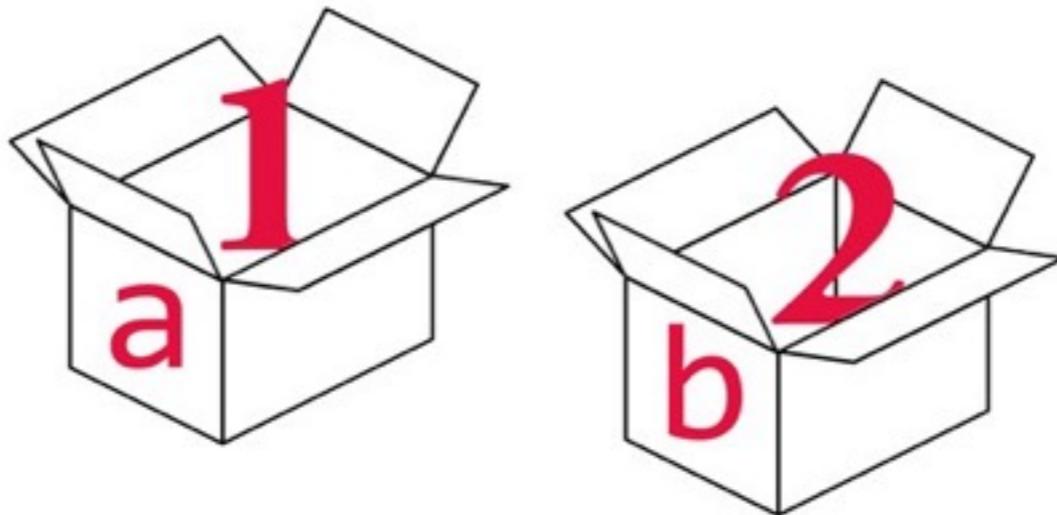
8 min

1. Open up sublime
2. Type in your pseudocode in the text editor

VARIABLES /
DATA TYPES

WHAT ARE VARIABLES?

- › We can tell our program to remember (store) values for us to use later on.
- › The 'container' we use to store the value is called a **variable**
- › A variable has a **name** and **value**.
- › That value can change.



DECLARING A VARIABLE

```
var age = 29;
```

JAVASCRIPT — UPDATING THE VALUE OF A VARIABLE

Declaring a variable:

```
var champion = "Sarah";
```

Update the value of the variable:

```
champion = "Bruce";
```

CREATING VARIABLES



DIRECTIONS

1. We'll be using the console to practice creating variables.
It's where JavaScript is interpreted and run. You can
use it to practice writing JavaScript!
2. Open up Google Chrome
3. Right click and go to "inspect"
4. Select "console"
5. Follow along!

VARIABLE PRACTICE

1. We'll be using the console to practice creating variables. It's where JavaScript is interpreted and run. You can use it to practice writing JavaScript!
2. Open up Google Chrome
3. Right click and go to "inspect"
4. Select "console"
5. Follow along!

VARIABLE PRACTICE

```
var name = "Susan";  
name;
```

```
var age = 18;  
age;
```

WHAT CAN BE STORED IN VARIABLES?

DATA TYPES:

STRINGS

"Today is Monday"

Letters and other
characters enclosed
in quotes

NUMBERS

10

22.75

- ▶ Positive numbers
- ▶ Negative numbers
- ▶ Decimals

BOOLEANS

true

false

- ▶ True
- ▶ False

VARIABLE PRACTICE

To predict the types of data you're dealing with you can use `typeof`.

```
typeof 3.45;  
typeof true;
```

JS BASICS

QUIZ

COMMON MISTAKES

"Bill" = var name;

COMMON MISTAKES

```
var name = "Bill";
```

COMMON MISTAKES

var total score = 20;

COMMON MISTAKES

```
var totalScore = 20;
```

COMMON MISTAKES

```
var fullName = Suzie Smith;
```

COMMON MISTAKES

```
var fullName = "Suzie Smith";
```

COMMON MISTAKES

Var fullName = "Bill Smith";

COMMON MISTAKES

```
var fullName = "Bill Smith";
```

COMMON MISTAKES

```
var score = "5";  
score += "6";
```

COMMON MISTAKES

```
var score = 5;  
score += 6;
```

ARITHMETIC // ASSIGNMENT OPERATORS

ARITHMETIC OPERATORS

NAME:

	OPERATOR:	EXAMPLE:	RESULT:
ADDITION	+	$2 + 4$	6
SUBTRACTION	-	$8 - 1$	7
MULTIPLICATION	*	$2 * 3$	6
DIVISION	/	$4 / 2$	2

ASSIGNMENT OPERATORS

	INITIAL VALUE:	OPERATOR:	EXAMPLE:	RESULT:
ASSIGN VALUE TO VARIABLE	var num = 8	=	num = 6	6

ASSIGNMENT OPERATORS

	INITIAL VALUE:	OPERATOR:	EXAMPLE:	RESULT:
ADD VALUE TO VARIABLE	var num = 8	+=	num += 6	14

ASSIGNMENT OPERATORS

	INITIAL VALUE:	OPERATOR:	EXAMPLE:	RESULT:
SUBTRACT VALUE FROM VARIABLE	var num = 8	-=	num -= 6	2

ASSIGNMENT OPERATORS

```
var totalAmount = 6;  
totalAmount += 4;  
totalAmount -= 2;
```

What will total amount be equal to?

ASSIGNMENT OPERATORS

```
var score = 6;  
score + 2;
```

What will score be equal to?

EXERCISE — VARIABLES



KEY OBJECTIVE

- ▶ Practice declaring and assigning variables

TYPE OF EXERCISE

- ▶ Individual/paired

LOCATION

- ▶ Starter Code > variables

EXECUTION

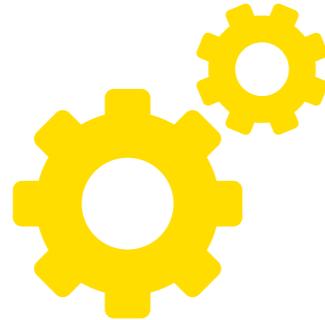
6 min

1. Follow the instructions under Part 1 & Part 2

INTRO TO JQUERY

YOUR RESPONSIBILITIES

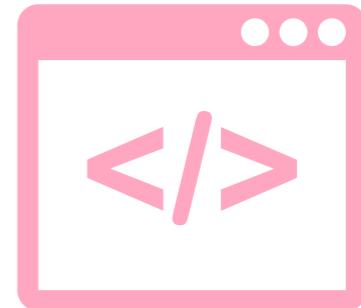
Don't feel like you have to sit down and memorize the syntax!



Focus on understanding
the key concepts



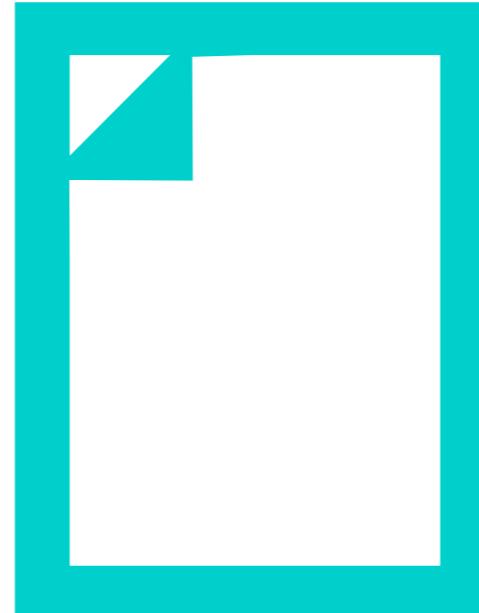
Be resourceful. Google
is your best friend



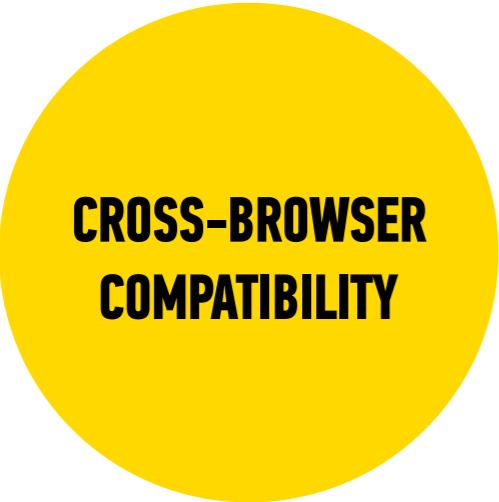
Practice, practice,
practice

INTRO TO JQUERY — YOUR NEW BEST FRIEND!

jQuery is a JavaScript [file](#) you include in your pages.



INTRO TO JQUERY — YOUR NEW BEST FRIEND!



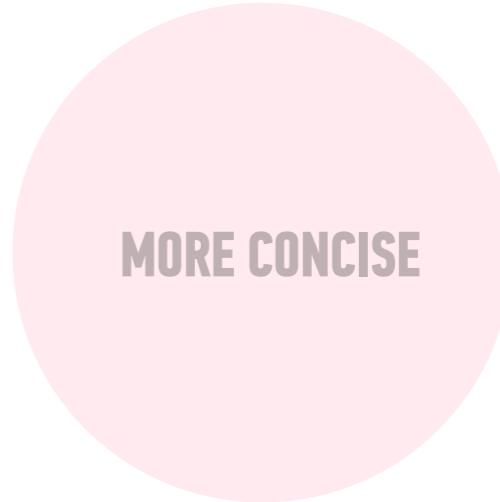
**CROSS-BROWSER
COMPATIBILITY**

Works the same
in all browsers



**FAMILIAR
SYNTAX**

Use more familiar,
CSS-style syntax



MORE CONCISE

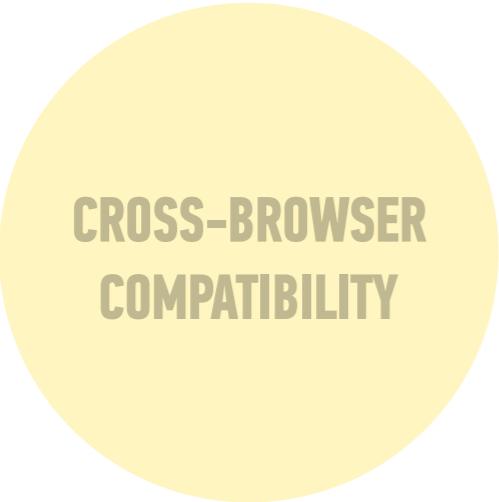
Write way less code to
achieve the same tasks

JQUERY VS. JAVASCRIPT

jQuery will ensure that our code works the same in different browsers.

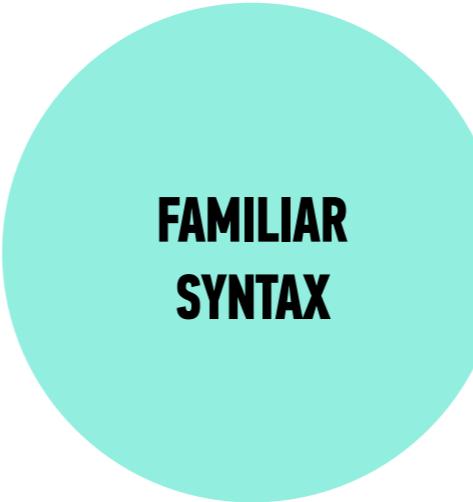


INTRO TO JQUERY — YOUR NEW BEST FRIEND!



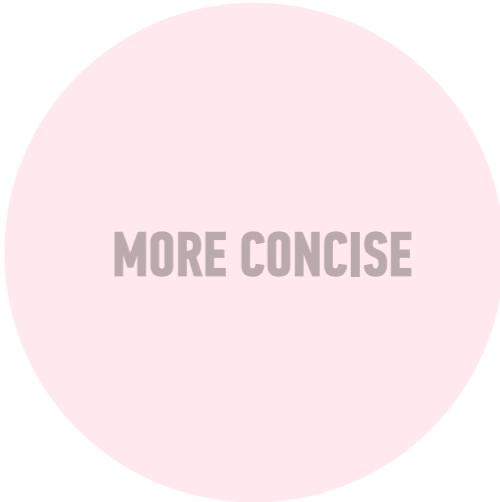
**CROSS-BROWSER
COMPATIBILITY**

Works the same
in all browsers



**FAMILIAR
SYNTAX**

Use more familiar,
CSS-style syntax



MORE CONCISE

Write way less code to
achieve the same tasks

JQUERY VS. JAVASCRIPT

jQuery allows us to use the CSS-style selectors that we know and love! Yay!

JS:



```
document.getElementsByTagName('body')[0]
```



```
document.getElementById('about')
```

JQUERY:

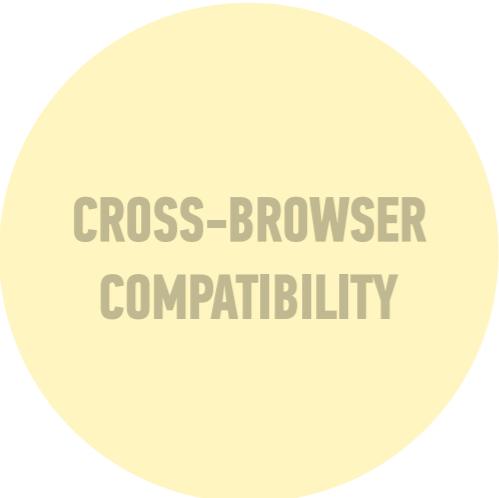


```
$( 'body' )
```



```
$( '#about' )
```

INTRO TO JQUERY — YOUR NEW BEST FRIEND!



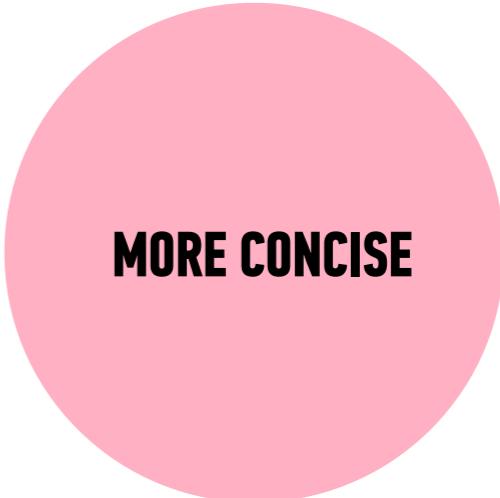
**CROSS-BROWSER
COMPATIBILITY**

Works the same
in all browsers



**FAMILIAR
SYNTAX**

Use more familiar,
CSS-style syntax



MORE CONCISE

Write way less code to
achieve the same tasks

JQUERY VS. JAVASCRIPT

JS:

```
document.getElementById('heading').innerHTML = "Your Name";
```



JQUERY:

```
$('#heading').html('Your Name');
```



You could do everything jQuery does with plain-old vanilla Javascript

JQUERY VS. JAVASCRIPT — A COMPARISON OF BENEFITS

JQUERY

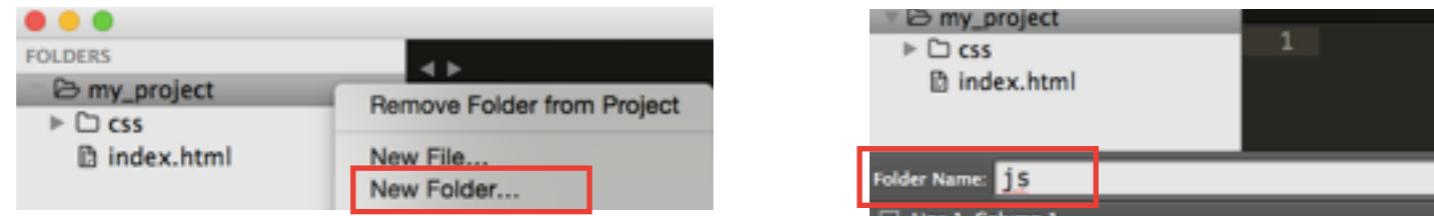
- ▶ Write way less code to achieve the same tasks
- ▶ Cross-browser compatibility
- ▶ Use more familiar, CSS-style syntax

PURE JAVASCRIPT

- ▶ Better performance
- ▶ Faster

STEP 1: ADD JQUERY TO YOUR WEBSITE

1. Download the [jQuery](#) script (version 3.x, compressed).
2. Add a js folder to your project



3. Move the jQuery file to the js folder

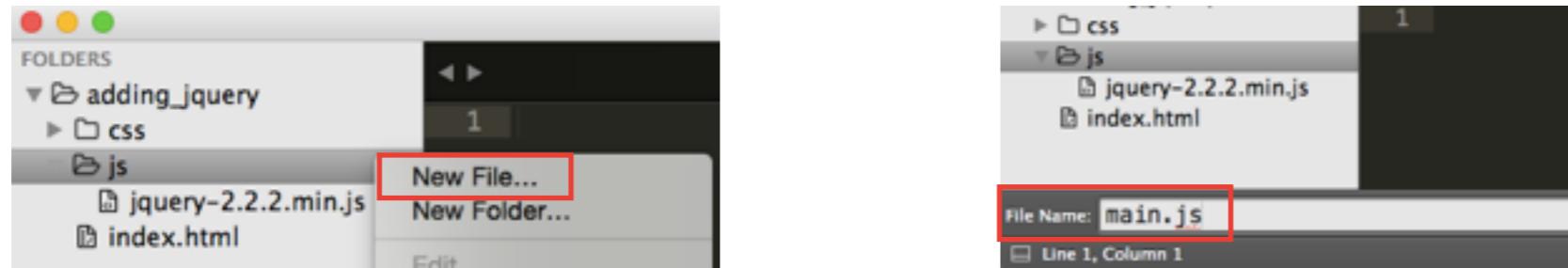


4. Use a <script> tag to include the jQuery file after your HTML content and before any other JavaScript files that use it.

```
<body>
  <!-- HTML content here -->
  <script src="js/jquery-1.11.2.min.js"></script>
  <!-- Javascript file will go here -->
</body>
```

STEP 2: ADD A JAVASCRIPT FILE

1. Create a Javascript file. This process will be similar to creating an HTML or CSS file, but this time the file should have a .js extension (example: main.js)



2. Link to the Javascript file from your HTML page using the <script> element. Add this **right before the closing body tag and after your jquery file**.

```
<body>
  <!-- HTML content here -->
  <script src="js/jquery-1.11.2.min.js"></script>
  <script src="js/main.js"></script>
</body>
```



ORDER IS IMPORTANT!!!!

using JQUERY

USING JQUERY TO MANIPULATE THE DOM

1

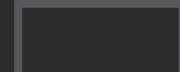
Select an element/elements

2

Work with those elements

JQUERY — SELECTING ELEMENTS

Selector



```
$('li').addClass('selected');
```

```
$('#info') = jquery('#info')
```

JQUERY OBJECTS — FINDING ELEMENTS: SOME EXAMPLES

	CSS	JQUERY
ELEMENT	<code>a { color: blue; }</code>	<code>\$(‘a’)</code>
ID	<code>#special { color: blue; }</code>	<code>\$(‘#special’)</code>
CLASS	<code>.info { color: blue; }</code>	<code>\$(‘.info’)</code>
NESTED SELECTOR	<code>div span { color: blue; }</code>	<code>\$(‘div span’)</code>

USING JQUERY TO MANIPULATE THE DOM

1

Select an element/elements

2

Work with those elements

JQUERY — WORKING WITH THOSE ELEMENTS

```
$('li').addClass('selected');
```

Parameter(s)

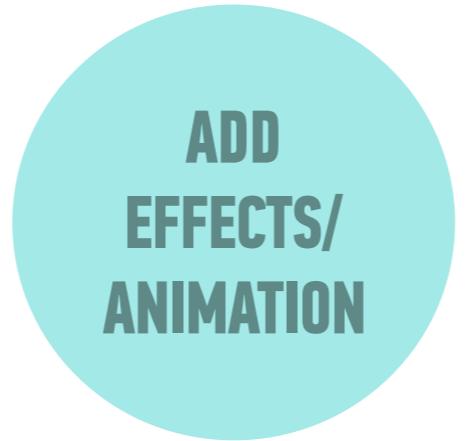
Method

JQUERY METHODS — WORKING WITH THOSE ELEMENTS

After we've selected elements, we can use jQuery methods to:



**GET/SET
CONTENT**



**ADD
EFFECTS/
ANIMATION**



**CREATE
EVENT
LISTENERS**



See your handout, pages 304-305 in the textbook, or the [jQuery docs](#) for list!

GET/SET CONTENT

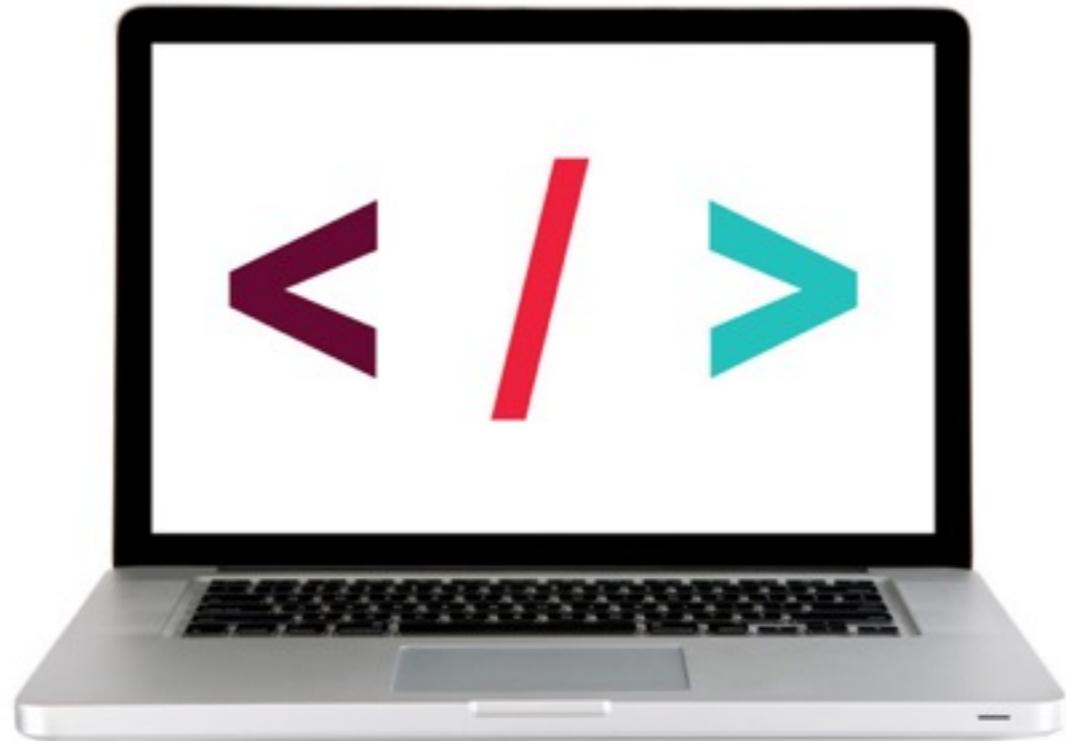
GETTING/SETTING CONTENT — PART 1

Get/change content of elements and attributes

METHODS	EXAMPLES
.html()	<code>\$('h1').html('Content to insert goes here');</code>
.attr()	<code>\$('img').attr('src', 'images/bike.png');</code>
.css()	<code>\$('#box1').css('color', 'red');</code>

What goes in the parentheses?
The **html or styles** you want to change.

LET'S TAKE A CLOSER LOOK



GET/SET CONTENT

GETTING/SETTING CONTENT — PART 2

Get/change content of elements and attributes

METHODS	EXAMPLES
.addClass()	<code>\$('p').addClass('success');</code>
.removeClass()	<code>\$('p').removeClass('my-class-here');</code>
.toggleClass()	<code>\$('p').toggleClass('special');</code>

What goes in the parentheses?
The **classes** you want to change.

JQUERY METHODS — GETTING/SETTING CONTENT

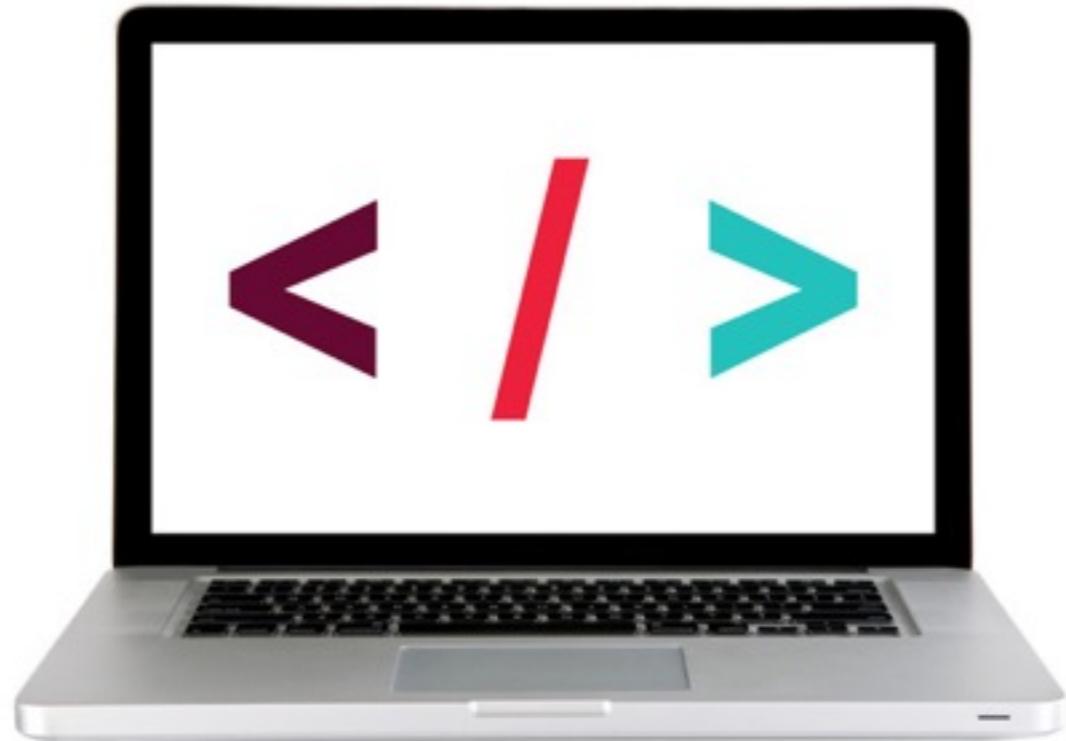
GET/SET
CONTENT

```
$('li').addClass('selected');
```



NO PERIOD!!!

LET'S TAKE A CLOSER LOOK



ACTIVITY



KEY OBJECTIVE

- ▶ Utilize jQuery to access and manipulate DOM elements.

TYPE OF EXERCISE

- ▶ Individual/Partner

AS A CLASS

5 min

Exercise is in starter code > jquery_code_along

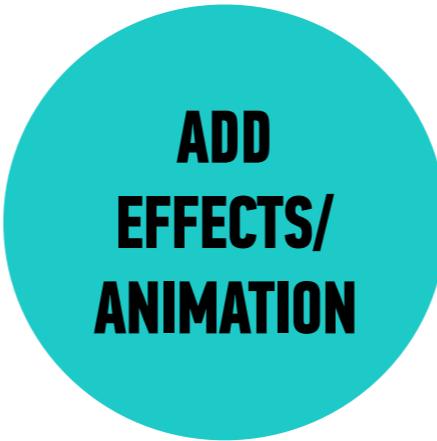
1. Follow the instructions under part 1 in main.js
2. Use slides as a guide for syntax

JQUERY METHODS — WORKING WITH THOSE ELEMENTS

After we've selected elements, we can use jQuery methods to:



**GET/SET
CONTENT**



**ADD
EFFECTS/
ANIMATION**



**CREATE
EVENT
LISTENERS**



See your handout, pages 304-305 in the textbook, or the [jQuery docs](#) for list!

**ADD
EFFECTS/
ANIMATION**

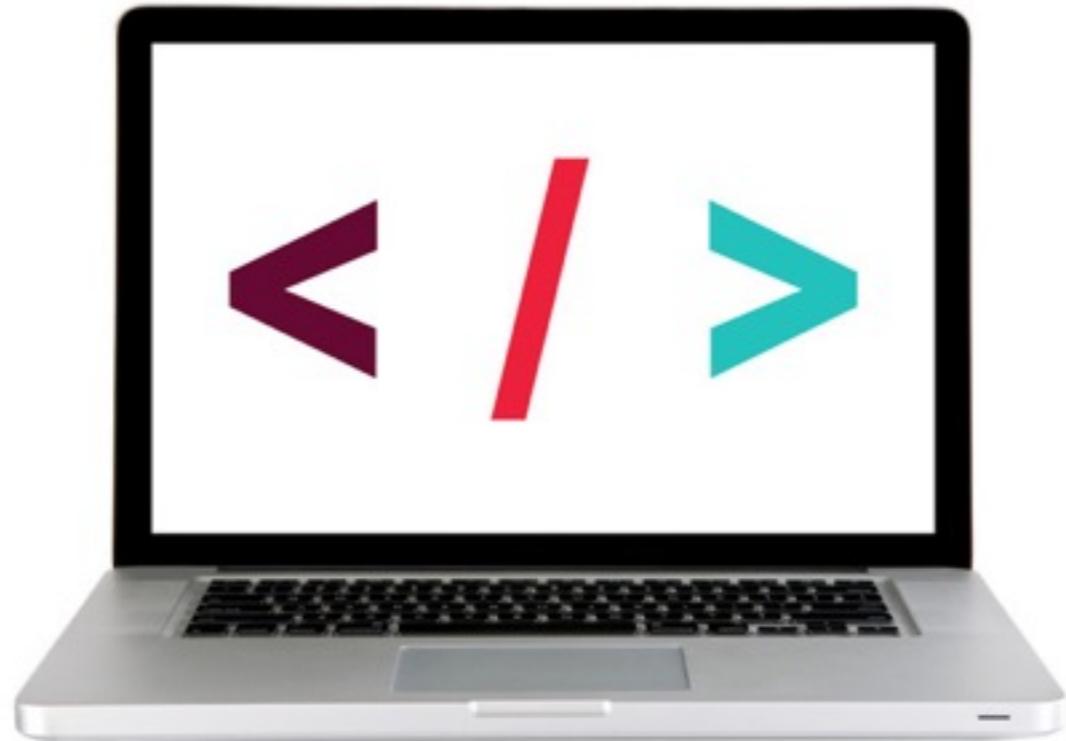
JQUERY METHODS — EFFECTS/ANIMATION

Add effects and animation to parts of the page

METHODS	EXAMPLES
.show()	<code>\$('h1').show();</code>
.hide()	<code>\$('ul').hide();</code>
.fadeIn()	<code>\$('h1').fadeIn(300);</code>
.fadeOut()	<code>\$('.special').fadeOut('fast');</code>
.slideUp()	<code>\$('div').slideUp();</code>
.slideDown()	<code>\$('#box1').slideDown('slow');</code>
.slideToggle()	<code>\$('p').slideToggle(300);</code>

What goes in the parenthesis?
An animation speed

LET'S TAKE A CLOSER LOOK

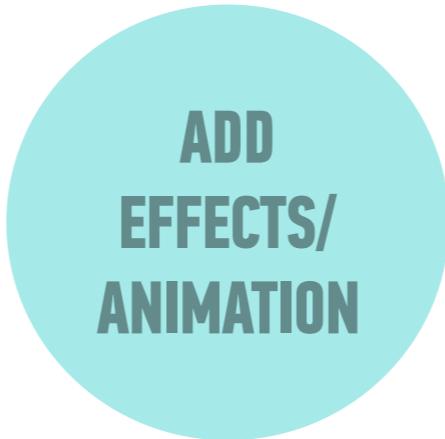


JQUERY METHODS — WORKING WITH THOSE ELEMENTS

After we've selected elements, we can use jQuery methods to:



**GET/SET
CONTENT**



**ADD
EFFECTS/
ANIMATION**

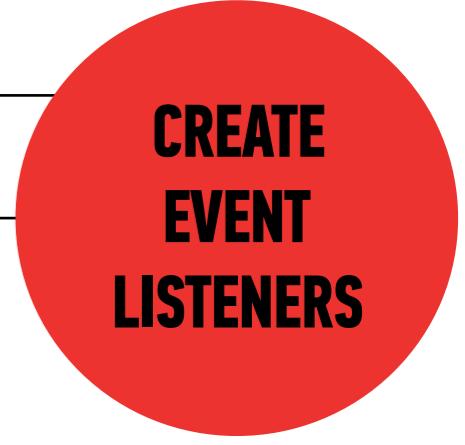


**CREATE
EVENT
LISTENERS**



See your handout, pages 304-305 in the textbook, or the [jQuery docs](#) for list!

JQUERY METHODS — EVENTS!



CREATE
EVENT
LISTENERS

We can use the `on()` method to handle all events in jQuery.

JQUERY METHODS — EVENTS!

CREATE
EVENT
LISTENERS

selector



`$('li')`

`// your code here`

`}`

JQUERY METHODS — EVENTS!

CREATE
EVENT
LISTENERS

method for all events

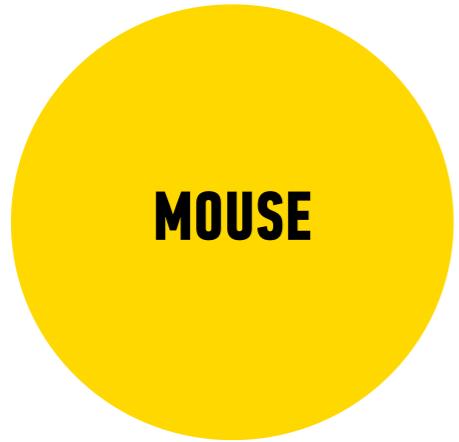
```
$           .on(  
    // your code here  
});
```

JQUERY METHODS — EVENTS!

CREATE
EVENT
LISTENERS

```
$ type of event
    // your code here
}
```

'click'

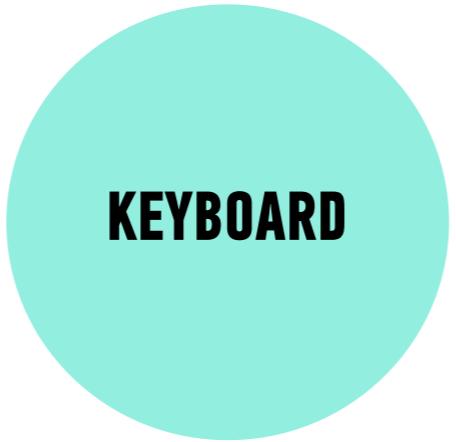


click

dblclick

mouseenter

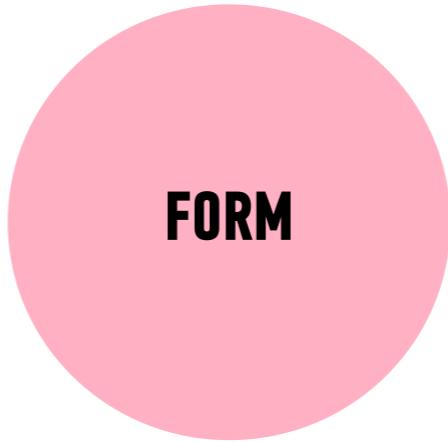
mouseleave



keypress

keydown

keyup



submit

change

focus

blur



resize

scroll

↓
\$('li').on('eventGoesHere'
 // your code here
});

JQUERY METHODS — EVENTS!

CREATE
EVENT
LISTENERS

```
$ , function() {  
    // your code here  
}
```

function to run
when event is
triggered

JQUERY METHODS — EVENTS!

CREATE
EVENT
LISTENERS

```
$('li').on('click', function() {  
    // your code here  
});
```

selector method for all events type of event

function to run when event is triggered

ACTIVITY



KEY OBJECTIVE

- ▶ Utilize jQuery to access and manipulate DOM elements.

TYPE OF EXERCISE

- ▶ Individual/Partner

AS A CLASS

5 min

Exercise is in starter code > jquery_code_along

1. Follow the instructions under Part 2 in main.js
2. Use slides as a guide for syntax

CONDITIONALS

JAVASCRIPT — COMPARISON OPERATORS

>= Greater than or equal to Equal to **==**

<= Less than or equal to Not equal to **!=**

> Greater than

< Less than

ASSIGNMENT VS. COMPARISON — DON'T GET THEM CONFUSED!

ASSIGNMENT



```
var number = 7;
```

COMPARISON



```
if (number === 8) {  
    // Do something  
}
```

JAVASCRIPT — IF/ELSE IF/ELSE

```
if (answer === 38) {  
    // Do something if first condition is true  
} else if (answer === 30) {  
    // Do something second condition is true  
} else {  
    // Do something if all above conditions are false  
}
```

LOGICAL OPERATORS

JAVASCRIPT — LOGICAL OPERATORS

&& and

|| or

! not

MULTIPLE CONDITIONS

```
if (name === "GA" && password === "YellowPencil"){
    //Allow access to dashboard
}
```

EXERCISE — VARIABLES



KEY OBJECTIVE

- ▶ Practice declaring and assigning variables

TYPE OF EXERCISE

- ▶ Individual/paired

LOCATION

- ▶ Starter Code > Conditionals

EXECUTION

6 min

1. Follow the instructions in main.js

HTML BASICS

LEARNING OBJECTIVES

- › Gain an overview of the JavaScript landscape and its placement in the web ecosystem.
- › Practice programmatic thinking by writing pseudo-code.
- › Write expressions that both assign and evaluate variables.
- › Explain the difference between jQuery and vanilla JS.
- › Register and trigger event handlers for jQuery events.

FUNCTIONS

KEEP LEARNING!!

KEEP GOING!!



**BUILD, BUILD,
BUILD!!!**



**GA WORKSHOPS
AND EVENTS**



**ONLINE
RESOURCES**

KEEP LEARNING AT GA

- ▶ Try your hand at the jquery_practice lab in your starter code (solution is also provided)
- ▶ Tackle the last two Dash projects which cover the material we went over tonight

KEEP GOING!!



**BUILD, BUILD,
BUILD!!!**

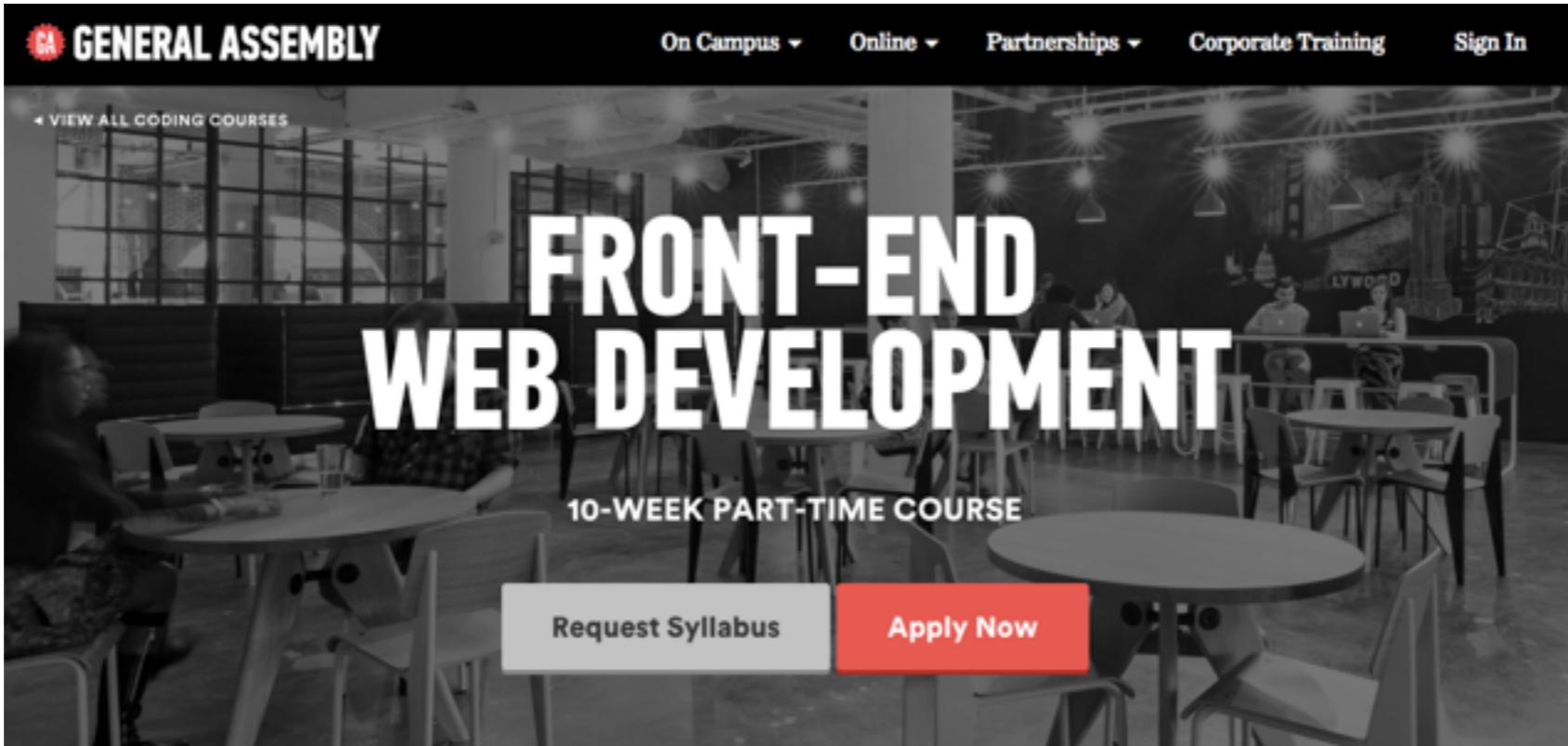


**GA WORKSHOPS
AND EVENTS**



**ONLINE
RESOURCES**

KEEP LEARNING AT GA



The image shows the landing page for General Assembly's Front-end Web Development course. The page features a black and white photograph of a modern classroom or workshop space with several round tables and students working on laptops. Overlaid on the image is the course title "FRONT-END WEB DEVELOPMENT" in large, bold, white letters, followed by "10-WEEK PART-TIME COURSE" in smaller white text. At the bottom of the main image are two buttons: "Request Syllabus" in a grey box and "Apply Now" in a red box. Above the main image, the General Assembly logo and navigation menu are visible.

GENERAL ASSEMBLY

On Campus ▾ Online ▾ Partnerships ▾ Corporate Training Sign In

VIEW ALL CODING COURSES

FRONT-END WEB DEVELOPMENT

10-WEEK PART-TIME COURSE

Request Syllabus Apply Now

Overview

Learning Support

Student Work

A circular icon containing a yellow gear with a white center, flanked by less than and greater than symbols (< >).

Skills & Tools

Use HTML5, CSS3, JavaScript, jQuery, GitHub, and Sublime in concert.

KEEP LEARNING AT GA

The screenshot shows the General Assembly website for a JavaScript Development course. At the top, there's a navigation bar with the GA logo, "GENERAL ASSEMBLY", "VIEW ALL CODING COURSES", and links for "On Campus", "Online", "For Companies", and "Sign In". The main hero image features a man working at a laptop. Overlaid text reads "JAVASCRIPT DEVELOPMENT" in large letters, "10-WEEK PART-TIME COURSE" below it, and two buttons: "Request Syllabus" (gray) and "Apply Now" (red). Below the hero image is a sidebar with links: "Overview" (selected), "Learning Support", "Curriculum", and "Tuition & Dates". To the right, there are two circular icons: one with a yellow gear and arrows labeled "Skills & Tools", and another with a yellow and gray bar and arrows labeled "Production Standard". Text next to these icons describes the course focus on learning JavaScript and building single-page web apps.

GENERAL ASSEMBLY

VIEW ALL CODING COURSES

On Campus ▾ Online ▾ For Companies ▾ Sign In

JAVASCRIPT DEVELOPMENT

10-WEEK PART-TIME COURSE

Request Syllabus Apply Now

Overview

Learning Support

Curriculum

Tuition & Dates

< >

Skills & Tools

Learn to code in JavaScript, the native language of the web used by developers the world over.

< >

Production Standard

Build a single-page web app that persists user data and connects to services like Twitter and Facebook via APIs.

KEEP GOING!!



**BUILD, BUILD,
BUILD!!!**



**GA WORKSHOPS
AND EVENTS**



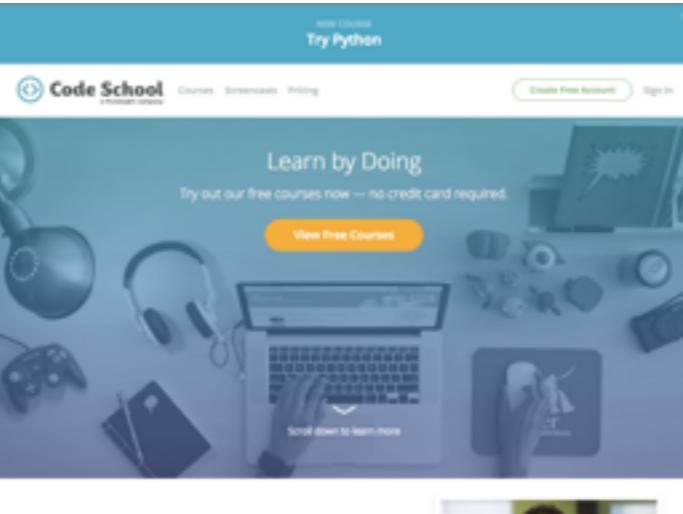
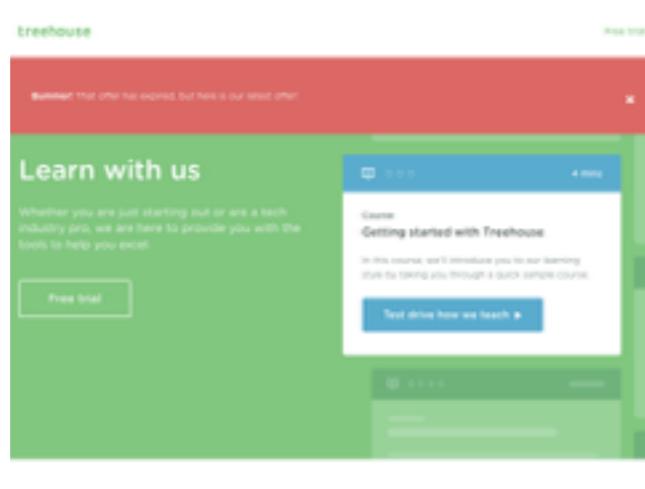
**ONLINE
RESOURCES**

KEEP LEARNING ONLINE

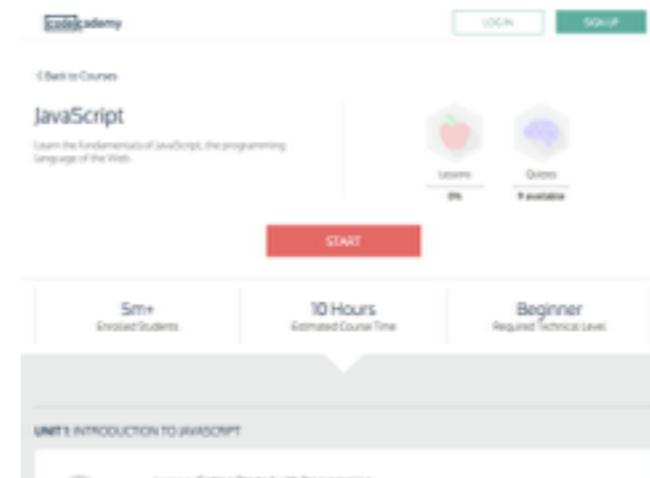
DASH (GA)



TEAM TREEHOUSE



CODE SCHOOL



CODEACADEMY

JAVASCRIPT 101

EXIT TICKETS