



---

**FRONT-END WEB DEVELOPMENT**

---

**SNACKS & DESIGN**

**TODAY!**

**LAURA**

**(GOOGLE SHEET IS PINNED IN SLACK)**

---

**FEWD**

---

**Q & A**

*“How is the code written after the pseudo is figured out?”*

*“Will we be going over jquery in more depth when we get to writting js?”*

*“I still need to see how jQuery is written into Javascript. Is it fair to say that scss::CSS is jQuery::Javascript?”*

---

# AGENDA

---



- Cont. Intro to Programming
- Intro to jQuery
- Using jQuery
- Method Chaining
- Debugging
- jQuery Documentation
- Lab

---

**FEWD**

---

# INTRO TO JS

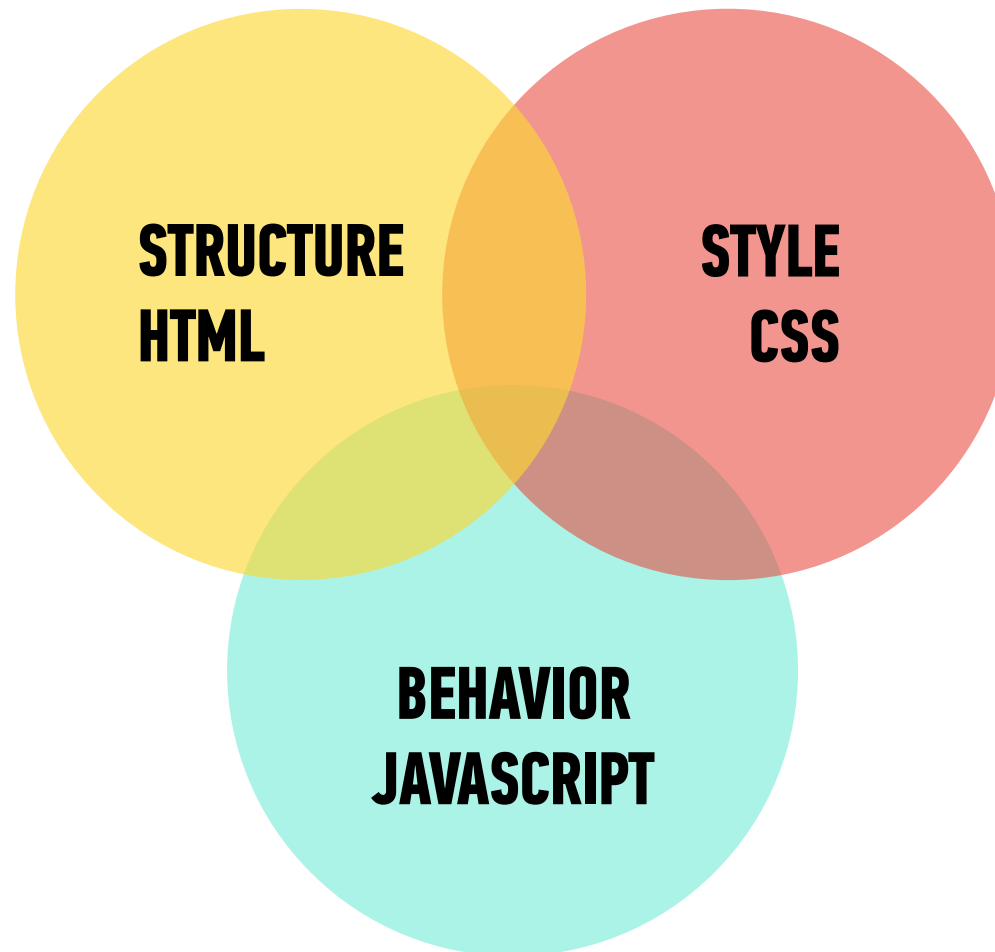


---

## THE THREE AMIGOS: STRUCTURE, STYLE, BEHAVIOR

---

- HTML = Noun
- CSS = Adjective
- Javascript = Verb



# JAVA VS. JAVASCRIPT

---

*Just a quick note! We're learning **JavaScript** in this class, not Java. Java and JavaScript are actually two different languages.*



!=



---

## 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:

- Select the text inside all the `<p>` elements on a page
- Select the element that has the id attribute with a value of **email**
- Find out what the user entered into a text input when they submit a form

---

## 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
- Add or remove a class from elements to trigger new CSS rules for those elements

---

## WHAT JAVASCRIPT CAN DO – MODIFYING CONTENT

---

Please Enter Your Details

REQUIRED

Some fields below require your attention

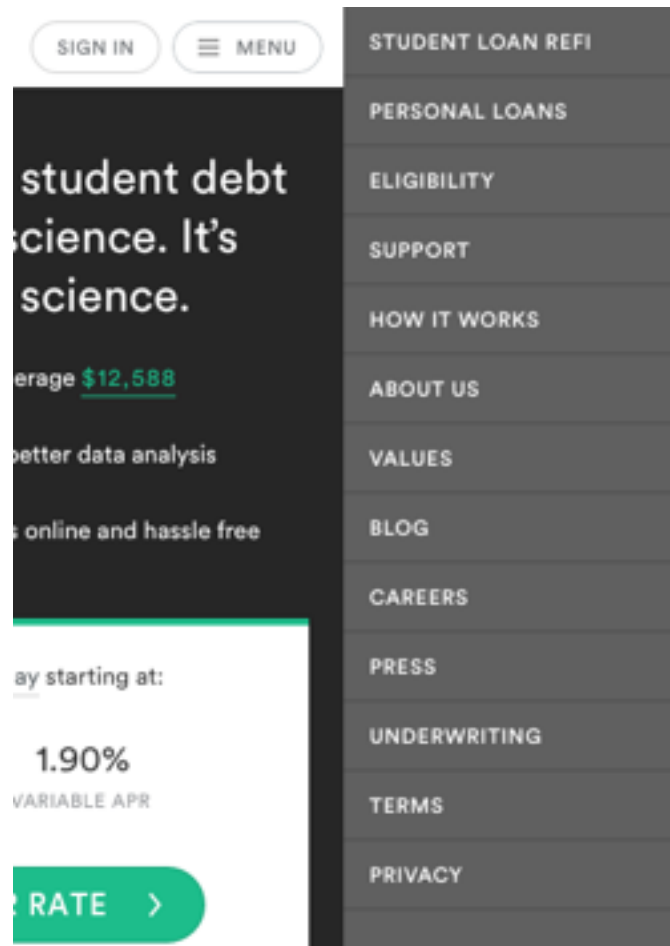
FIRST NAME	Sarah
LAST NAME	Holden
HIGHEST DEGREE	Choose One
SCHOOL	Enter School Name
EMPLOYER	Enter Employer Name
JOB TITLE	Enter Job Title
ANNUAL INCOME	\$ Enter Your Annual Income (Not Household)
BANKING + INVESTMENT TOTAL	\$ Enter Estimated Total
STUDENT LOAN BALANCE	\$ Approximate Amount (\$5,000 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 as the user scrolls down the page
- Check to make sure the user has entered a valid email address into a form and display an error message if not
- Open a chat panel when the user clicks on a 'Chat with Us' button
- Filter data when the user selects a filter



# WHAT JAVASCRIPT CAN DO – PROGRAM RULES

Filters

Clear All

Filter

Clear All

Cuisine

Clear

☐ Dinner • 28

☐ Lunch • 19

☐ Asian • 7

☐ Sandwiches • 7

☐ Healthy • 6

☐ Mediterranean • 6

☒ Thai • 5

☐ Vegetarian • 5

☐ Japanese • 4

☐ Latin American • 4

+ See all

Rating


★ ★ ★ ★ ★

& up

Pick A Restaurant

7 Restaurants nearby

Sort By Default



Trike Thai Noodles & Sushi

Japanese, Asian

★★★★★

75 Ratings

\$20


Min

\$3

Delivery

60-70 m

Est. Wait



New China Chinese Restaurant

Asian, Dinner

★★★★★

222 Ratings

\$15


Min

\$3 - \$5

Delivery

60-70 m

Est. Wait



Dante's Pizzeria (MILWAUKEE)

Dinner, Sandwiches

★★★★★

40 Ratings

\$18

Min

\$3

Delivery

90-100 m

Est. Wait

Filter data when the user selects a filter

---

## 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
- When a page has finished loading
- When the user hits enter to submit a form

GET YOUR RATE >

# JQUERY

*Eric Boyer*

# LEARNING OBJECTIVES

- Differentiate between jQuery and JavaScript, describe benefits of using them.
- Recognize jQuery syntax
- Use selectors and jQuery functions to effectively manipulate the DOM.

---

**JQUERY**

---

# INTRO TO JQUERY

---

## BECOMING A PROGRAMMER

---

*It isn't about the programming language!!!  
It is about changing how you think.*

---

## YOUR RESPONSIBILITIES

---

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

It's important that you:

- Focus on understanding the key concepts
- Are resourceful — we'll work on honing your Google-ing skills
- Get as much practice in as possible

---

**JQUERY**

---

# THE BASICS



---

# INTRO TO JQUERY — YOUR NEW BEST FRIEND!

---

## WHAT IS JQUERY?

- jQuery is a JavaScript file you include in your pages.
- Makes it faster and easier to write cross-browser JavaScript
- “*Cross browser*” - works the same in all\* browsers.
- Allows us to find elements using CSS-style selectors and then do something to them using jQuery methods
- Your new best friend!



---

## 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')
```



---

## 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**

---

# ADDING JQUERY TO YOUR PROJECT

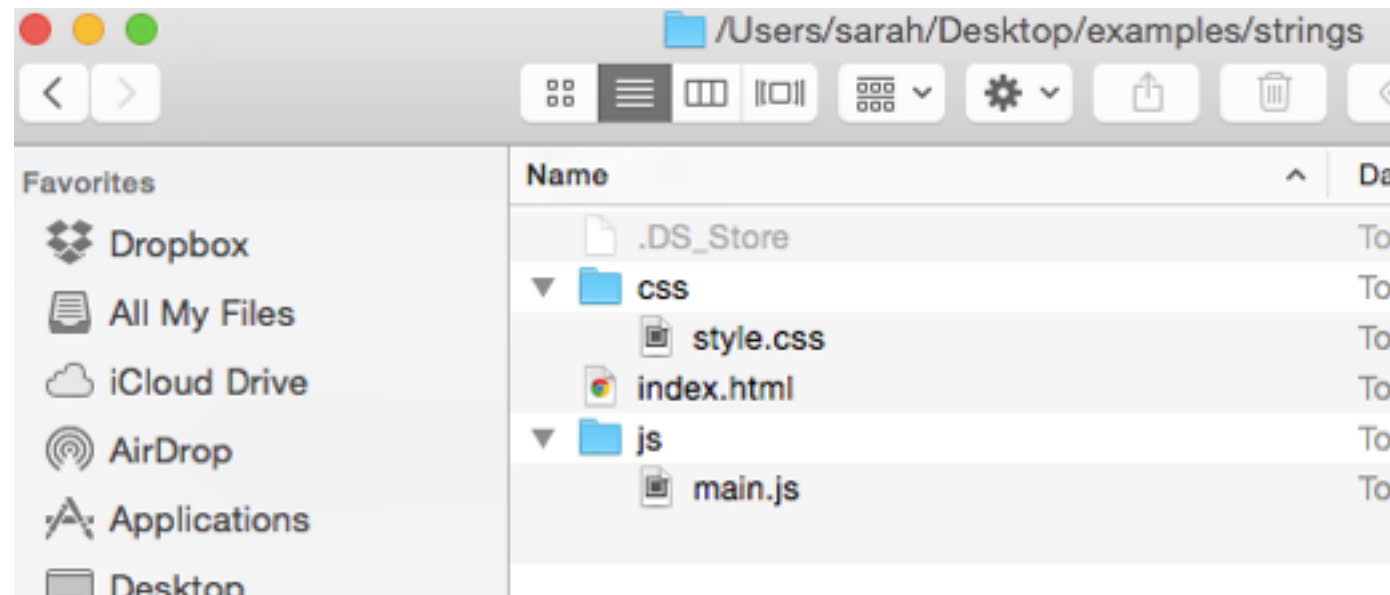
---

## KEEP IT ON THE UP AND UP!

---

- ▶ It is considered **best practice** to keep Javascript files organized in one folder.
- ▶ Usually people name this folder 'scripts', 'js', or 'javascript'.
- ▶ Use an underscore or dash between words in folder names instead of a space. And try to avoid characters/symbols in file names (*really\_cool\_page.html* or *really-cool-page.html*).

sample structure:



---

## STEP 1: ADD JQUERY TO YOUR WEBSITE

---

1. Download the [jQuery](#) script (version 2.x) and include it in your project. Keep things organized by placing it within your js folder.
2. Include jQuery in your HTML page before the closing `</body>` tag by adding a `<script>` element with a `src` that points to the jQuery file
3. Make sure to include jQuery **before** any other js 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. We'll almost always want to add this script element **right before the closing body tag**.

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



**ORDER IS IMPORTANT!!!!**

---

## PRO TIPS

---

### TO MAKE SURE YOUR JS IS HOOKED UP PROPERLY:

- Add an alert to the top of your JS file and load the page in the browser

```
alert('Hello from JS!');
```

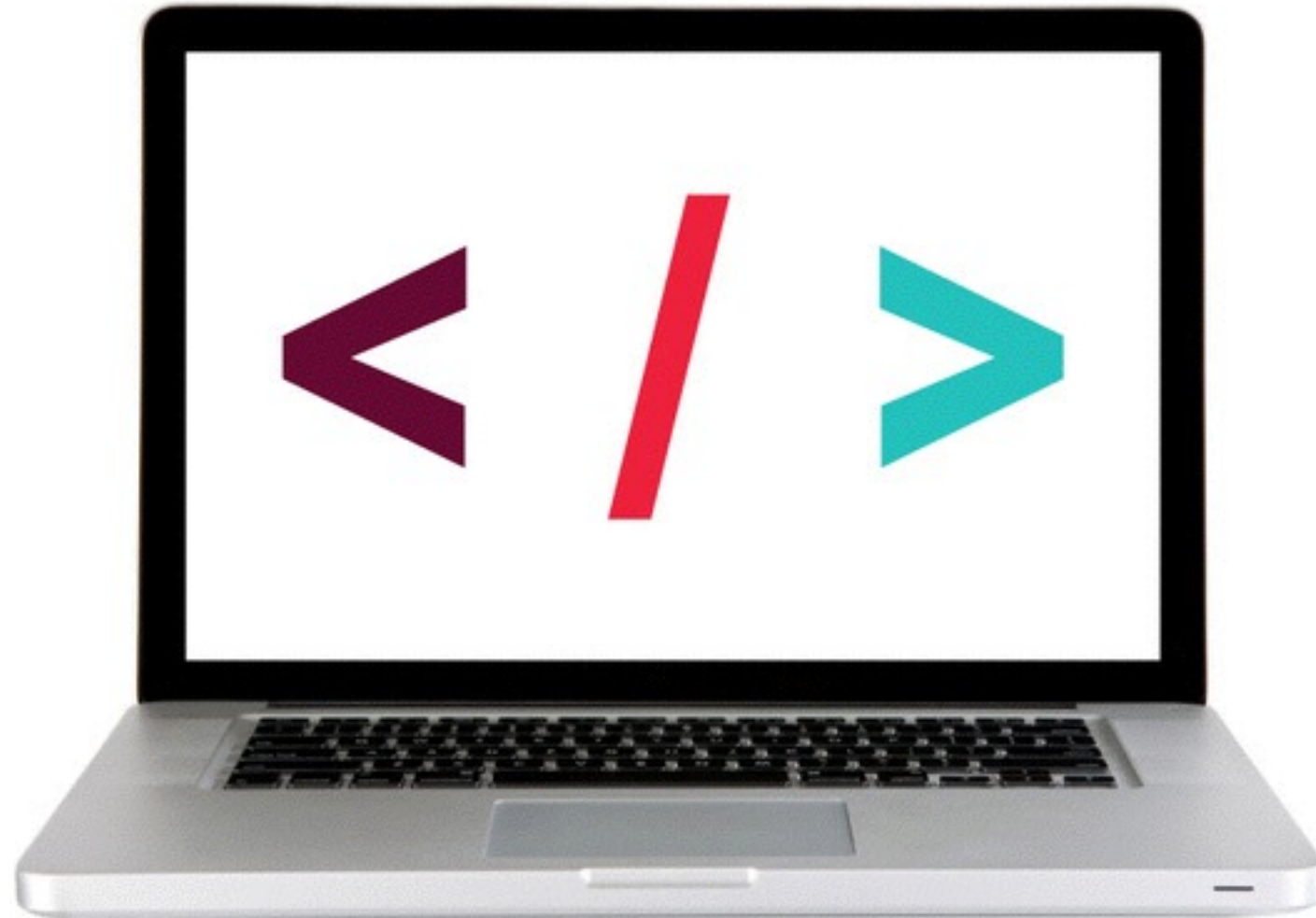
- If you don't see an alert pop up when you load the page, you know you have linked your files incorrectly.



---

**LET'S TAKE A CLOSER LOOK**

---



---

**JQUERY**

---

# USING JQUERY

---

## STATEMENTS

---

- Last class (and this) we are chatting about how scripts are a series of instructions that are executed one-by-one
- Each individual step is called a **statement**

```
var today = new Date();  
var hourNow = today.getHours()  
var greeting;  
  
if (hourNow > 18) {  
    greeting = 'Good evening';  
} else {  
    greeting = 'Good afternoon';  
}  
  
document.write(greeting);
```



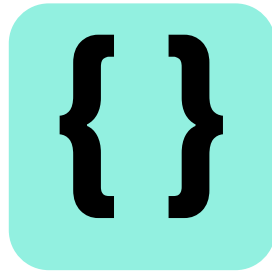
1. Begin on a new line
2. End with a **semicolon**

## JS SYNTAX

---

**Syntax:** Spelling and grammar rules of a programming language.

- Like any language, there are formal rules around how to write Javascript. This is the syntax.



---

## COMMENTS

---

```
// this is a single line comment
```

```
/*  
this  
is  
a  
multiline comment  
*/
```

Sublime shortcut: 1) Highlight what you want to comment 2) command + /

---

**JQUERY**

---

# **PART 1 — SELECT AN ELEMENT**

---

## USING JQUERY TO MANIPULATE THE DOM

---

**1**

Select an element/elements

**2**

Work with those elements

---

## JQUERY — SELECTING ELEMENTS

---

Selector

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

jQuery Function

### jQuery Function:

- ▶ Lets us find one or more elements in the page
- ▶ Creates a *jQuery object* which holds references to those elements
- ▶ We'll be using the shorthand in this class: `$()`
- ▶ `$(selector)` is the same as `jQuery(selector)`



---

## JQUERY OBJECTS — FINDING ELEMENTS: SOME EXAMPLES

---

- ▶ You can use your CSS-style selectors!!!

SELECTOR-		CSS:	JQUERY:
	CLASS	.className	\$('.className')
	ID	#idName	\$('#idName')
	MULTIPLE SELECTORS	h1, h2, h3	\$( 'h1, h2, h3' )
	DESCENDANT	li a	\$( 'li a' )

*& tons more!!!*

---

**JQUERY**

---

# **PART 2 — ADD A METHOD**

---

## USING JQUERY TO MANIPULATE THE DOM

---

**1**

Select an element/elements

**2**

Work with those elements

---

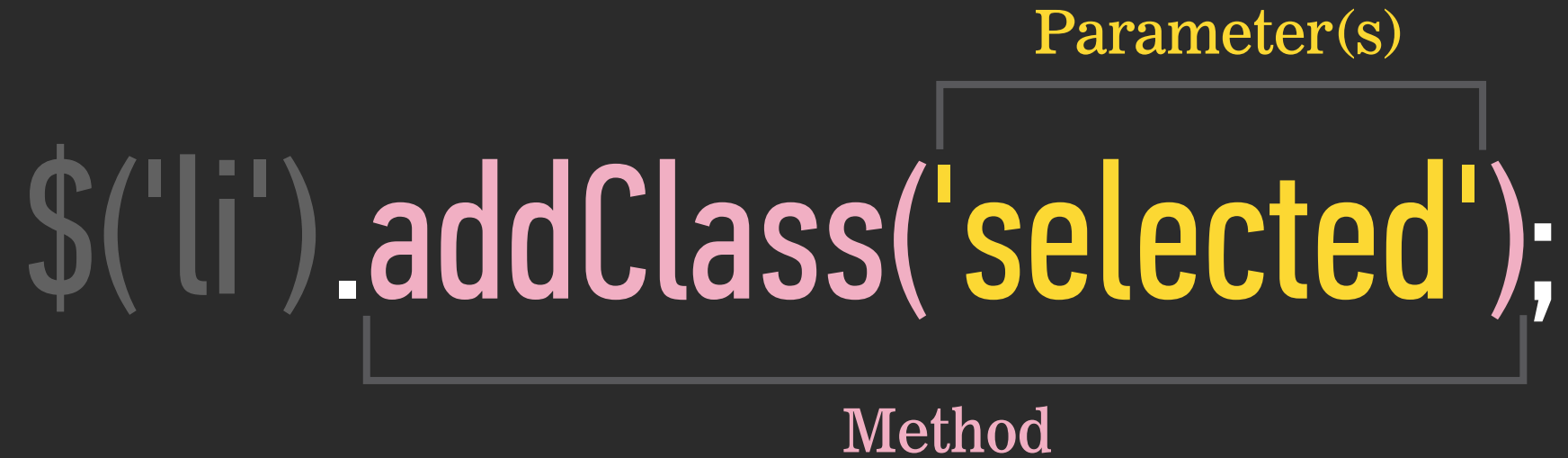
## JQUERY — WORKING WITH THOSE ELEMENTS

---

Parameter(s)

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

Method

The image shows the jQuery code snippet `$('li').addClass('selected');` with two annotations. A bracket above the string `'selected'` is labeled "Parameter(s)" in yellow text. A bracket below the `addClass` method name is labeled "Method" in pink text. The `addClass` text is pink, and the `'selected'` text is yellow.

---

## JQUERY METHODS

---

**Be forewarned!**

**There are a lot of methods!**

*Do not feel like you need to sit down and memorize these.* The important things is knowing that they're there and **being able to look them up** in the documentation.



See the [jQuery docs](#) for list!

---

## JQUERY METHODS — WORKING WITH THOSE ELEMENTS

---

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

**FINDING  
ELEMENTS**

**GET/SET  
CONTENT**

**ADD  
EFFECTS/  
ANIMATION**

**CREATE  
EVENT  
LISTENERS**

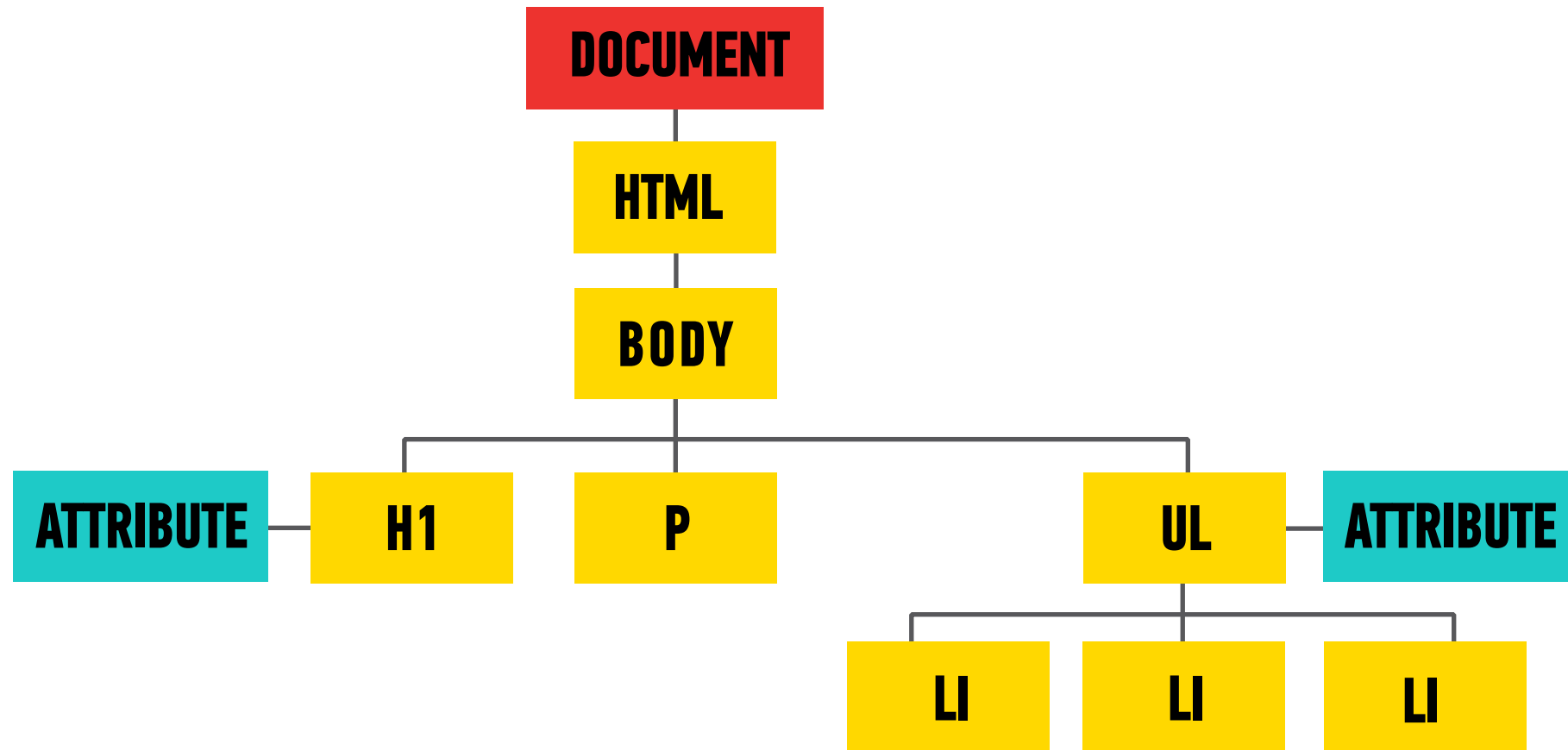


See the [jQuery docs](#) for list!

---

## TRAVERSING THE DOM?

---



## JQUERY METHODS — TRAVERSING THE DOM

### TRAVERSE THE DOM

jQuery provides us with methods to find/select elements to work with & traverse the DOM

Some methods available to us:

- ▶ `.find()` (finds all descendants)
- ▶ `.children()`
- ▶ `.parent()`
- ▶ `.siblings()`

What goes in the parentheses?  
A css-style **selector**

Examples:

```
$('h1').find('a');  
$('#box1').parent();  
$('p').siblings('.important');
```

*\*Think of this as part of the selection process, must come directly after another selection*



---

## JQUERY METHODS — WORKING WITH THOSE ELEMENTS

---

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

**FINDING  
ELEMENTS**

**GET/SET  
CONTENT**

**ADD  
EFFECTS/  
ANIMATION**

**CREATE  
EVENT  
LISTENERS**



See the [jQuery docs](#) for list!

## JQUERY METHODS — GETTING/SETTING CONTENT

### GET/SET CONTENT

Get/change content of elements, attributes, text nodes

Some methods available to us:

- ▶ `.html()`
- ▶ `.attr()`
- ▶ `.css()`
- ▶ `.addClass()`
- ▶ `.removeClass()`
- ▶ `.toggleClass()`

What goes in the parentheses?

The **html, styles, classes** you want to **add/change**

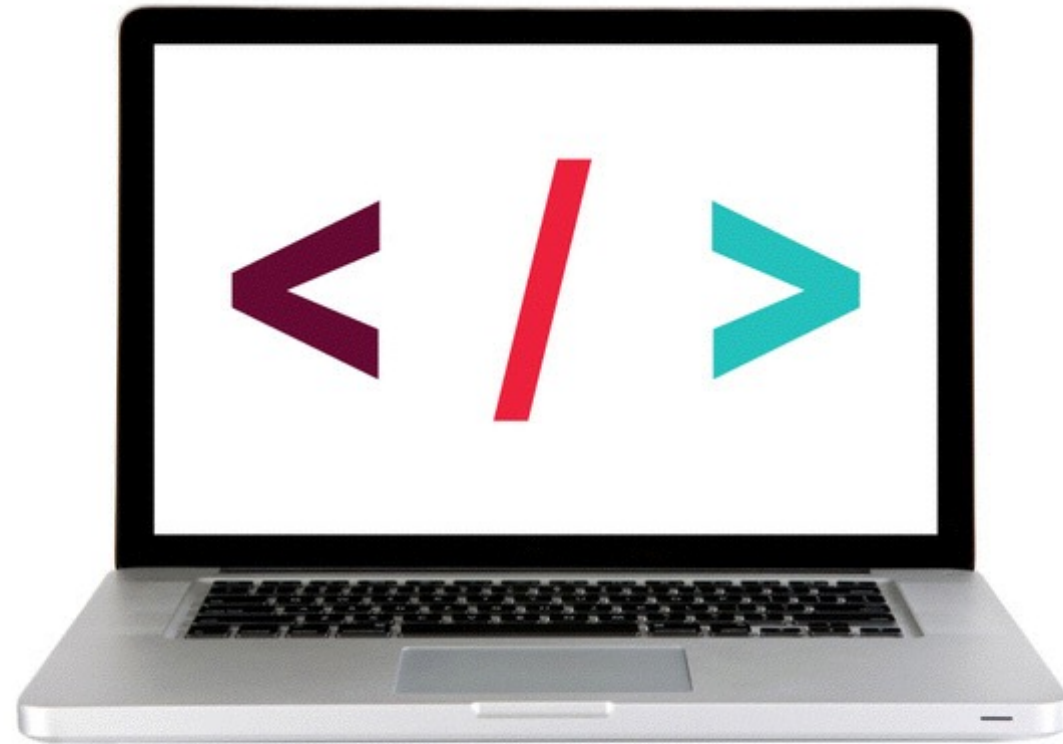
Examples of **adding/changing** content:

```
$('h1').html('Content to insert goes here');  
$('img').attr('src', 'images/bike.png');  
$('#box1').css('color', 'red');  
$('p').addClass('success');  
$('p').removeClass('my-class-here');
```

---

## LET'S TAKE A CLOSER LOOK

---



---

**JQUERY**

---

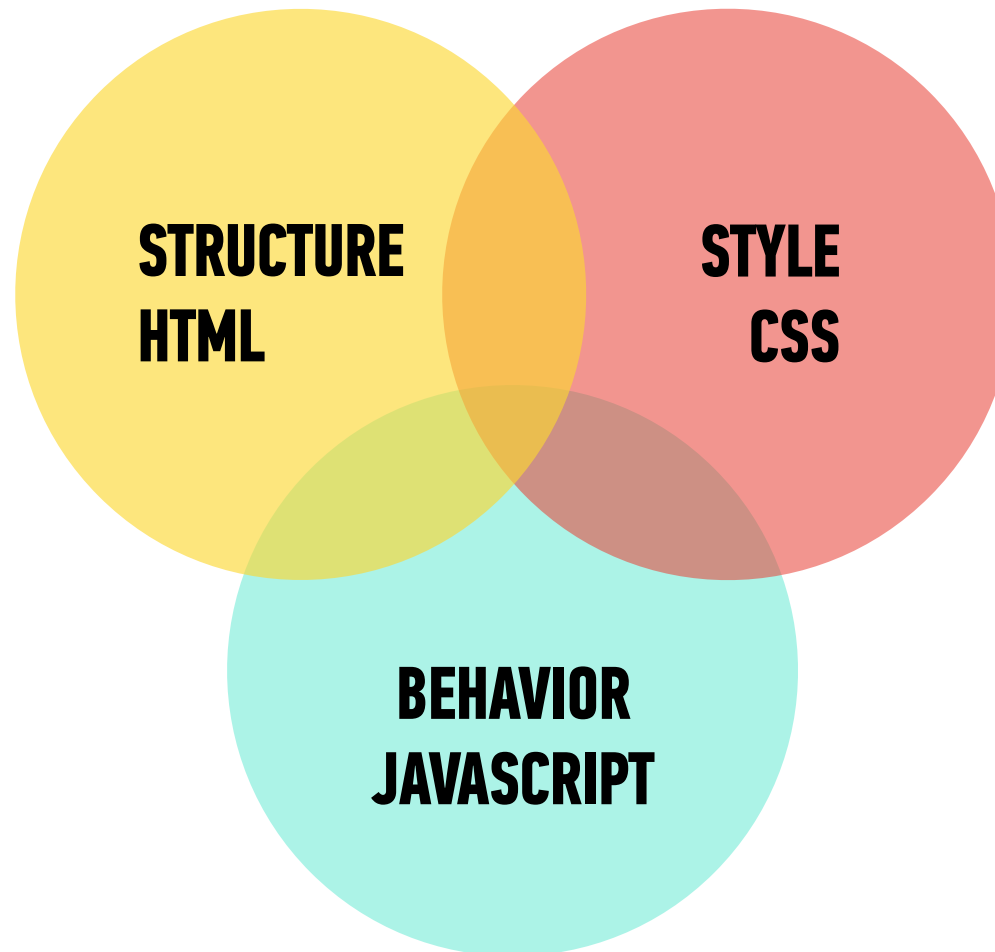
# SEPARATION OF CONCERNS

---

## THE THREE AMIGOS: STRUCTURE, STYLE, BEHAVIOR

---

- HTML = Noun
- CSS = Adjective
- Javascript = Verb

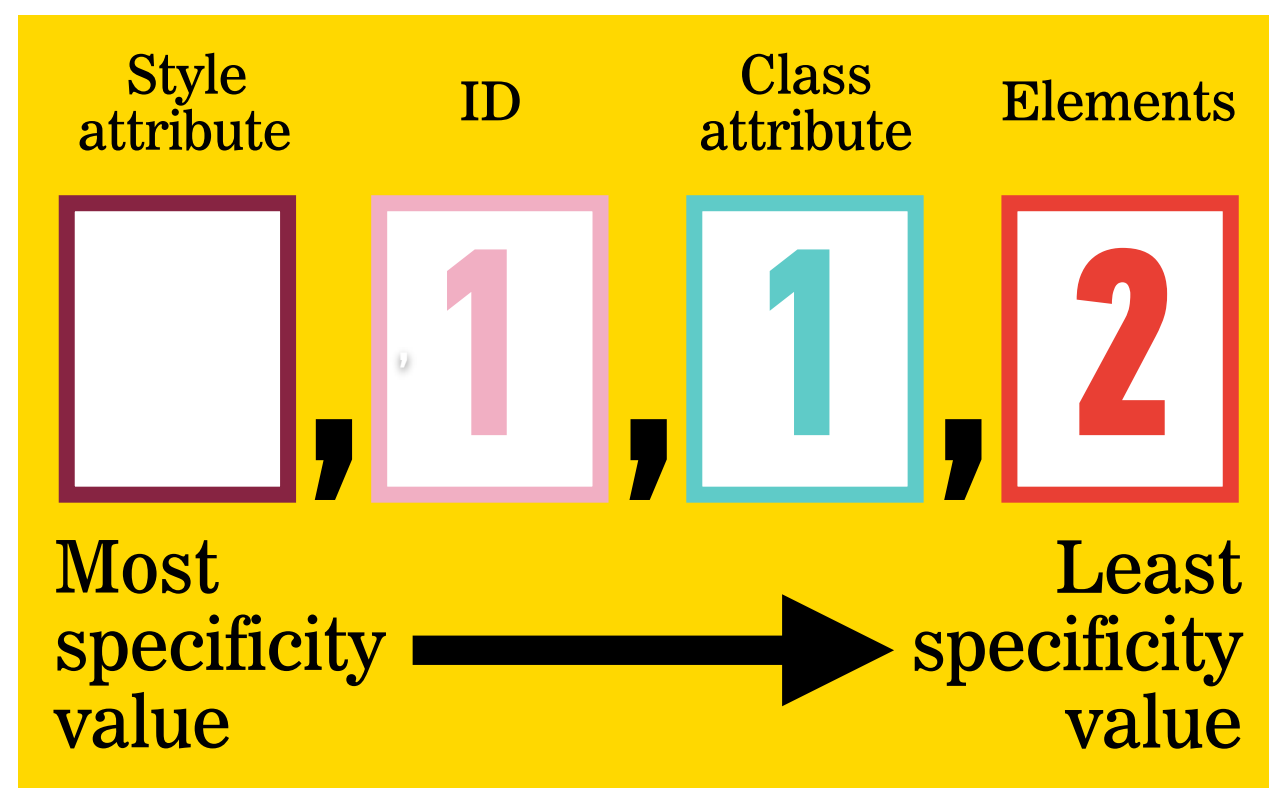
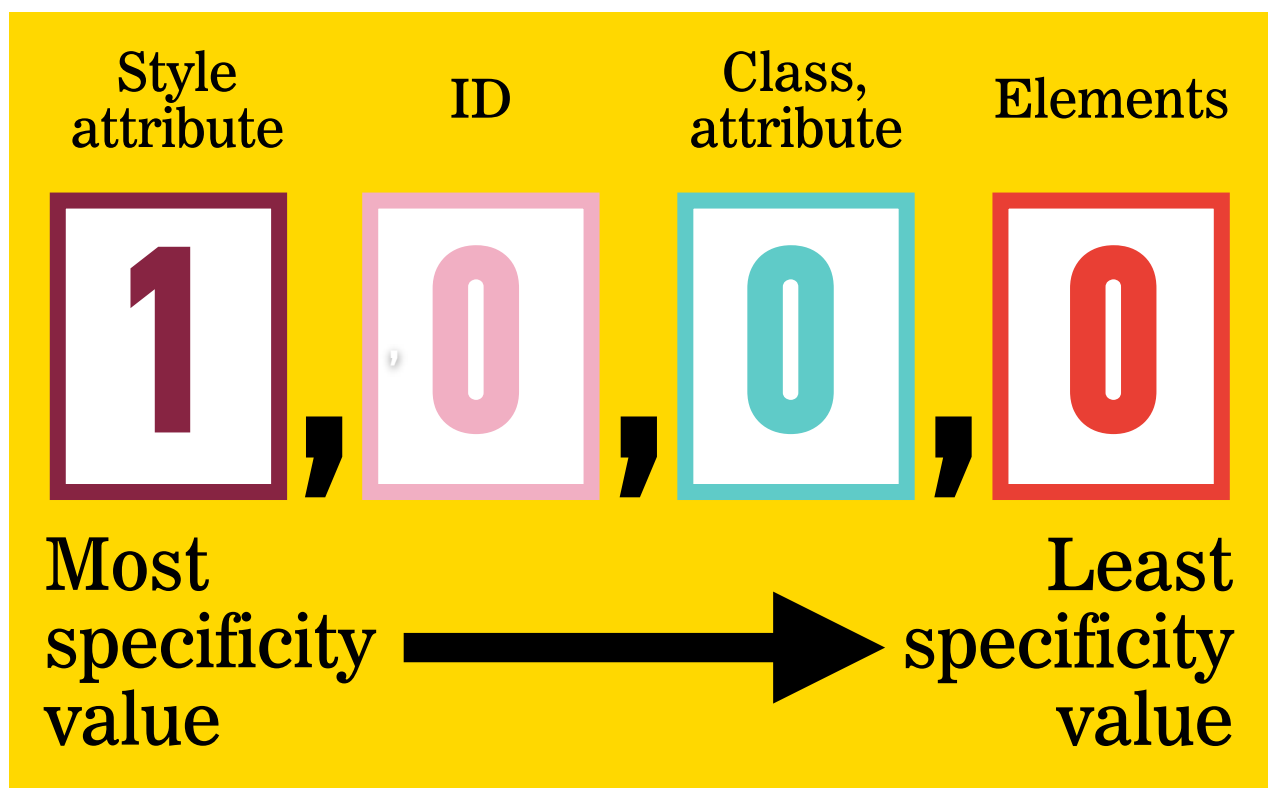


## CSS CASCADING

**WINNER!**

`<li style="color: red"></li>`

`#about .first h2 + li`



---

## SEPARATION OF CONCERNS

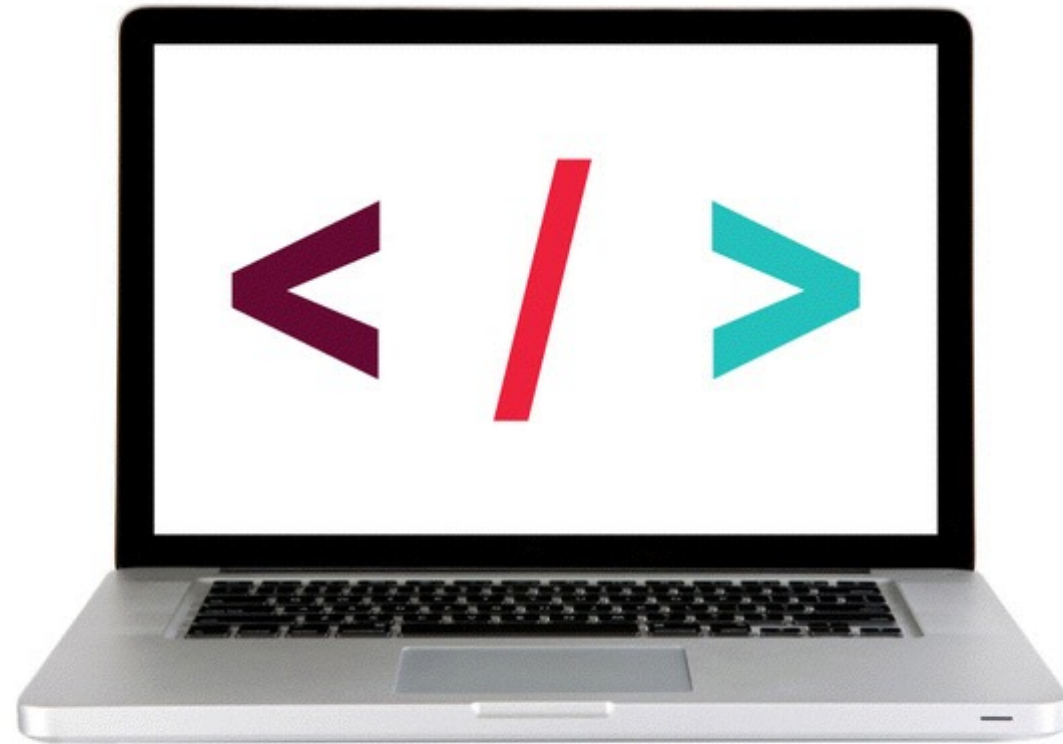
---

- Our JavaScript should focus on *behavior* and not on *presentation* (that's what our CSS is for!)
- How could we refactor our Color Switch from earlier to follow this guideline?
- (Fork this codepen)
- <http://codepen.io/eboyer/pen/grPZay>

---

## LET'S TAKE A CLOSER LOOK

---





---

## JQUERY METHODS — WORKING WITH THOSE ELEMENTS

---

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

**FINDING  
ELEMENTS**

**GET/SET  
CONTENT**

**ADD  
EFFECTS/  
ANIMATION**

**CREATE  
EVENT  
LISTENERS**



See the [jQuery docs](#) for list!

## JQUERY METHODS — EFFECTS/ANIMATION

**ADD  
EFFECTS/  
ANIMATION**

Add effects and animation to parts of the page

Some methods available to us:

- ▶ `.show()`
- ▶ `.hide()`
- ▶ `.fadeIn()`
- ▶ `.fadeOut()`
- ▶ `.slideUp()`
- ▶ `.slideDown()`
- ▶ `.slideToggle()`

What goes in the parenthesis?  
An **animation speed**

Examples:

```
$('h1').fadeOut(200);  
$('#box1').slideDown('slow');  
$('h1').fadeIn();
```

---

## JQUERY METHODS — WORKING WITH THOSE ELEMENTS

---

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

**FINDING  
ELEMENTS**

**GET/SET  
CONTENT**

**ADD  
EFFECTS/  
ANIMATION**

**CREATE  
EVENT  
LISTENERS**



See the [jQuery docs](#) for list!

## JQUERY METHODS — EVENTS!

---

**CREATE  
EVENT  
LISTENERS**

The `.on()` method is used to handle all events.

**Syntax:** `$('selector').on('event', code_that_should_run);`

**Example:**

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

## JQUERY METHODS — EVENTS!

CREATE  
EVENT  
LISTENERS

Some events that `.on()` deals with:

- ▶ **UI:** focus, blur, change
- ▶ **Keyboard:** keydown, keyup
- ▶ **Mouse:** click, mouseup, mousedown, mouseover
- ▶ **Form:** submit
- ▶ **Browser:** resize, scroll

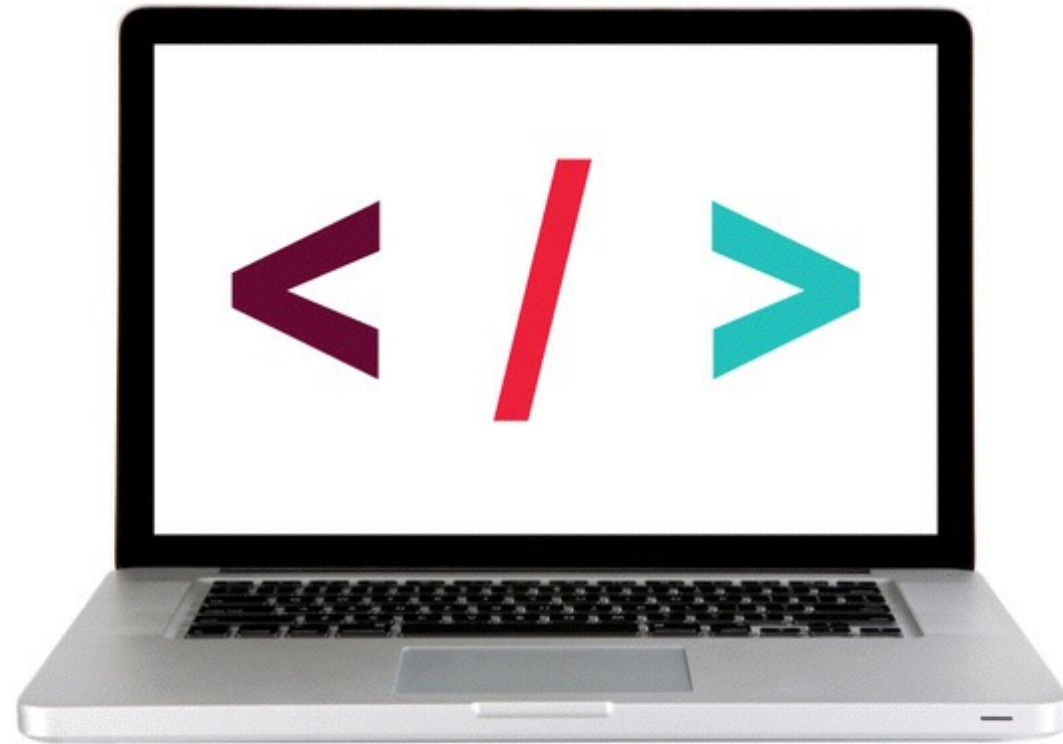


```
$('.li').on('eventGoesHere', function() {  
    // your code here  
});
```

---

## LET'S TAKE A CLOSER LOOK

---



---

**JQUERY**

---

# METHOD CHAINING

---

## JQUERY METHODS — METHOD CHAINING

---

```
$('.item').find('h2').slideUp(300)
```



---

## JQUERY METHODS — METHOD CHAINING

---

```
$('#main').children('p').fadeOut('slow')
```

---

## JQUERY METHODS — METHOD CHAINING

---

```
$('#h3').siblings('p').show()
```

---

**JQUERY**

---

**DEBUGGING**



**WHY ISN'T IT WORKING?**

---

## DEBUGGING — WHERE TO START

---

*Always start by defining the problem.*



**THE IMAGE IS NOT MOVING**



**NONE OF MY CODE WORKS**

---

## DEBUGGING — WHERE TO START

---

This will tell you where to start your hunt.



**THE IMAGE IS NOT MOVING**

*Find the code that makes  
the image move*



**NONE OF MY CODE WORKS**

*\* Syntax error, check console*

---

## DEBUGGING

---

*To access debugging console:*

Open the Chrome Inspector, click the Console tab

---

## DEBUGGING — LEVEL 1

---

### Check for errors in console

- The location may not be correct but is a good place to start.
- Ex: Unbalanced brackets or parentheses



Uncaught SyntaxError: Unexpected token )

main.js:13





---

## DEBUGGING — LEVEL 2

---

▸ `console.log()` can be used to display variable values in the console. This is useful for debugging.

```
console.log($('h2'));
```

This should print the element to the console.  
If it doesn't, there's probably something wrong with your selector.

*\*Shortcut to access console: cmd + opt + j*

---

## **DEBUGGING — LEVEL 3**

---

### **Do some Googling!**

- ▶ Try Googling it
- ▶ Be ready to clearly articulate the problem (Write out what your problem is)

---

## DEBUGGING — LEVEL 4

---

### Get help!

If you still can't find a solution, ask your fellow classmates and instructors.

Help us help you!

1. Zip what you're working on.
2. Post in Slack in the class channel.
3. Be descriptive about the problem.
4. Tell us what you've already done to try to figure it out.

---

**JQUERY**

---

# JQUERY DOCUMENTATION

---

## JQUERY DOCUMENTATION – IT'S YOUR FRIEND!

---

*Help! There's too much to learn! I feel overwhelmed!*

**A good developer is one that can look things up!!!**

1. The [jQuery Examples](#) page has an example for almost every method!
2. [jQuery documentation](#) — Look things up

---

## DASH – MAKING YOUR LIFE EASIER ONE DOC AT A TIME

---

I highly recommend that you download [Dash](#)



---

**JQUERY**

---

**LAB**

---

**LAB**

---





# ACTIVITY

---



## EXERCISE

### KEY OBJECTIVE

---

- ▶ Utilize jQuery tree traversal techniques to access and manipulate DOM elements.

### TYPE OF EXERCISE

---

- ▶ Partners/small groups

### AS A CLASS

---

- Until 8:50*      Lab is in lesson8\_starter\_code > [2] - select\_fun
1. Follow the instructions in main.js
  2. Use cheat sheet as a guide for syntax and look up any methods you're not familiar with in the jQuery documentation.
  3. **Bonus:** Complete part 2 of the lab for more practice (lesson8\_starter\_code > [3] - select\_fun\_part\_2)

# LEARNING OBJECTIVES

- Differentiate between jQuery and JavaScript, describe benefits of using them.
- Recognize jQuery syntax
- Use selectors and jQuery functions to effectively manipulate the DOM.

---

**LAB**

---

# **HOMEWORK**

---

## **HOMEWORK**

---

### **IF YOU ARE STUCK...**

- 1) Use the “Chrome Inspector” to look at your code, use the console, Google for answers.
  - 2) Ask your question in the Slack channel and see if any fellow students might know the answer
  - 3) Ask Eric
- \* When using your fellow students and instructors, pushing your code to Github is a great place to share where all of your code is currently at.

---

## **HOMEWORK**

---

### **DUE 6PM, SUNDAY**

- Complete the “select\_fun\_part\_2” lab assignment

---

**FRONT-END WEB DEVELOPMENT**

---

**SNACKS & DESIGN**

**MONDAY**

**FRANK**

**(GOOGLE SHEET IS PINNED IN SLACK)**

---

**JQUERY**

---

# EXIT TICKETS