# LET'S GET EVERYTHING SET UP!

1. Navigate to the FEWD 42 Dashboard (sarahеholden.com/fewd_dashboard/) and download the Lesson 8 starter code and slides. You'll want to keep the dashboard open for other links and resources we'll be referencing in class.
2. Move the starter code and slides from your Downloads folder to the **fewd** folder on your desktop.
3. Double-click on starter_code_lesson_8.zip to unzip it
4. After you've unzipped, be sure to delete the original .zip file!
5. Open the entire **fewd** folder with Sublime Text (either drag and drop the folder on the Sublime icon in the dock on Mac, or open Sublime and go to file > open… and select the **fewd** folder.
6. Log in to the FEWD 42 Slack (fewd42.slack.com) and join the class8 channel.

# WEEKLY OVERVIEW

| | |
|---|---|
| WEEK 4 | Intro to Programming / Intro to jQuery |
| WEEK 5 | Intro to JavaScript — Variables, Conditionals, Functions |
| WEEK 6 | JavaScript Continued — Arrays / Lab |

GENERAL ASSEMBLY

# JQUERY

*Sarah Holden*

# 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.

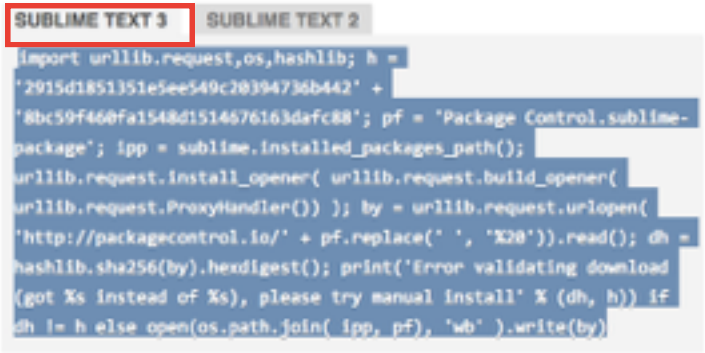
# AGENDA

- Intro to jQuery
- Using jQuery
- Method Chaining
- Debugging
- jQuery Documentation
- Lab

# LET'S INSTALL PACKAGE CONTROL!!

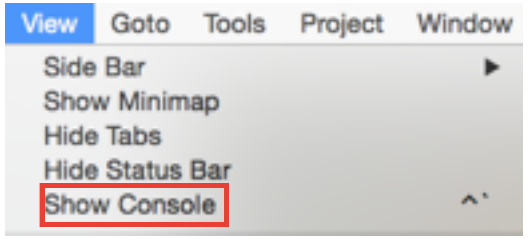
- Visit the Package Control site
- Click on "Install"

- Select and copy the code in the Sublime Text 3 tab

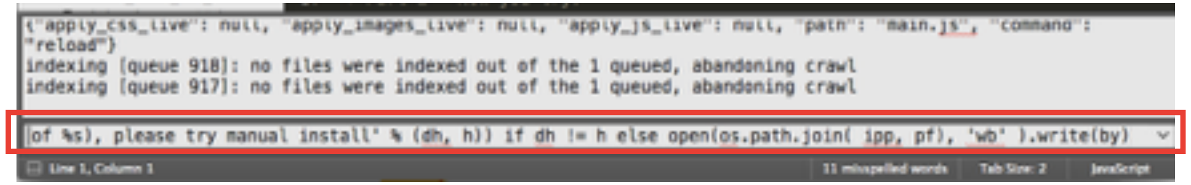


```
import urllib.request,os,hashlib; h =
'2915d1851351e5ee549c20394736b442' +
'8bc59f460fa1548d1514676163dafc88'; pf = 'Package Control.sublime-
package'; ipp = sublime.installed_packages_path();
urllib.request.install_opener( urllib.request.build_opener(
urllib.request.ProxyHandler()) ); by = urllib.request.urlopen(
'http://packagecontrol.io/' + pf.replace(' ', '%20')).read(); dh =
hashlib.sha256(by).hexdigest(); print('Error validating download
(got %s instead of %s), please try manual install' % (dh, h)) if
dh != h else open(os.path.join( ipp, pf), 'wb' ).write(by)
```

- In Sublime Text go to View > Show Console



- Paste the code in the bottom input field and hit the enter key

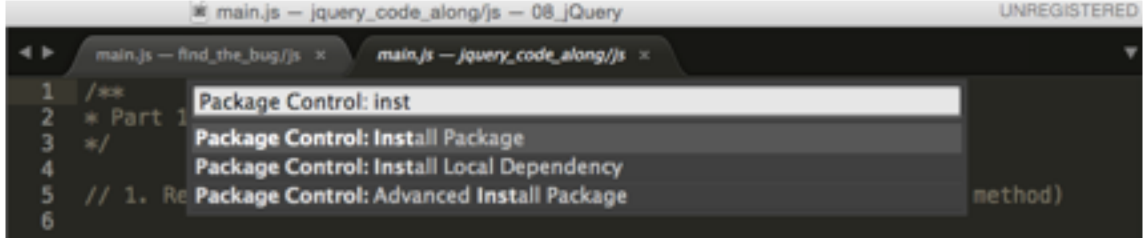


```
{"apply_css_live": null, "apply_images_live": null, "apply_js_live": null, "path": "main.js", "command":
"reload"}
indexing [queue 918]: no files were indexed out of the 1 queued, abandoning crawl
indexing [queue 917]: no files were indexed out of the 1 queued, abandoning crawl
```

```
of %s), please try manual install' % (dh, h)) if dh != h else open(os.path.join( ipp, pf), 'wb' ).write(by)
```

# LET'S INSTALL PACKAGE CONTROL!!

- To install packages go to Sublime Text > Preferences > Package Control (or cmd + shift + p)
- When a popup opens, type "Install Package" and select that option

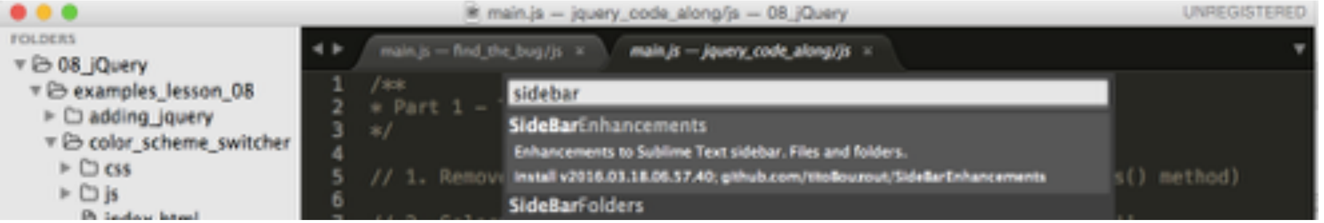


Package Control: inst

- Package Control: Install Package
- Package Control: Install Local Dependency
- Package Control: Advanced Install Package

main.js — jquery_code_along/js — 08_jQuery
UNREGISTERED
main.js — find_the_bug/js
main.js — jquery_code_along/js
/**
* Part 1
*/
// 1. Re
method)

- Search for "sidebarEnhancements"



sidebar

- SideBarEnhancements
  Enhancements to Sublime Text sidebar. Files and folders.
  install v2016.03.18.06.57.40; github.com/titoBouzout/SideBarEnhancements
- SideBarFolders

- Click on "sidebarEnhancements" to install that package.

# 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

# THE BASICS

# INTRO TO JQUERY — YOUR NEW BEST FRIEND!

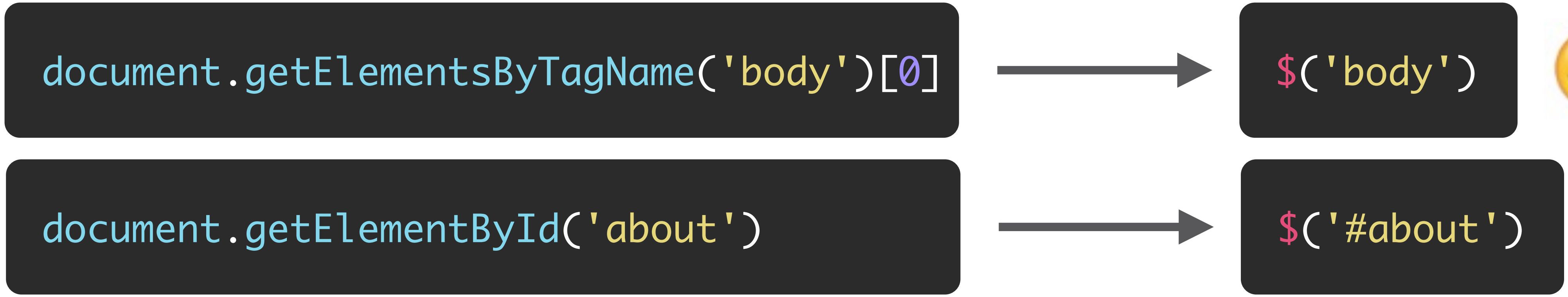
## WHAT IS JQUERY?

- 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: JQUERY:

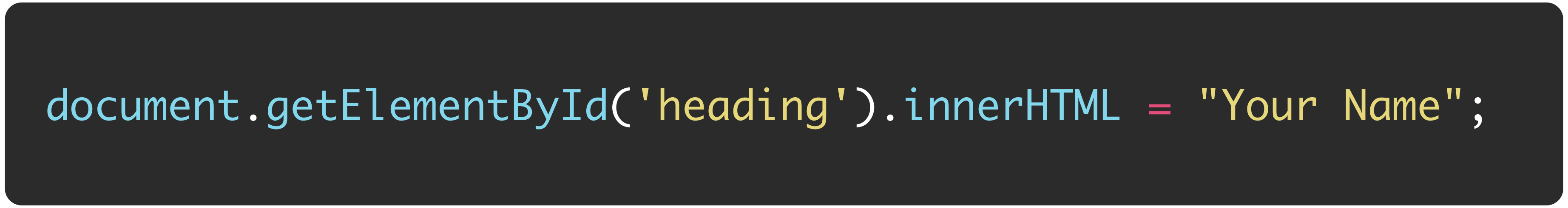


😭 `document.getElementsByTagName('body')[0]` → `$('body')` 😀

😭 `document.getElementById('about')` → `$('#about')` 😀

# JQUERY VS. JAVASCRIPT

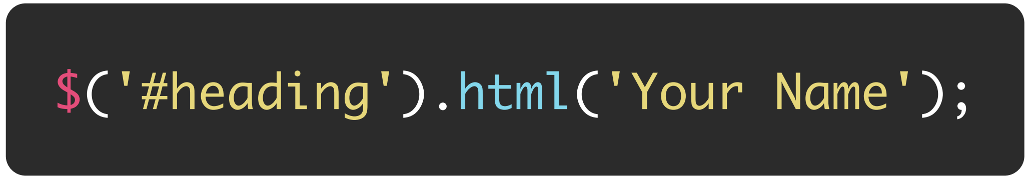
**JS:**



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

😭

**JQUERY:**



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

😀

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

# JQUERY VS. JAVASCRIPT — A COMPARISON OF BENEFITS

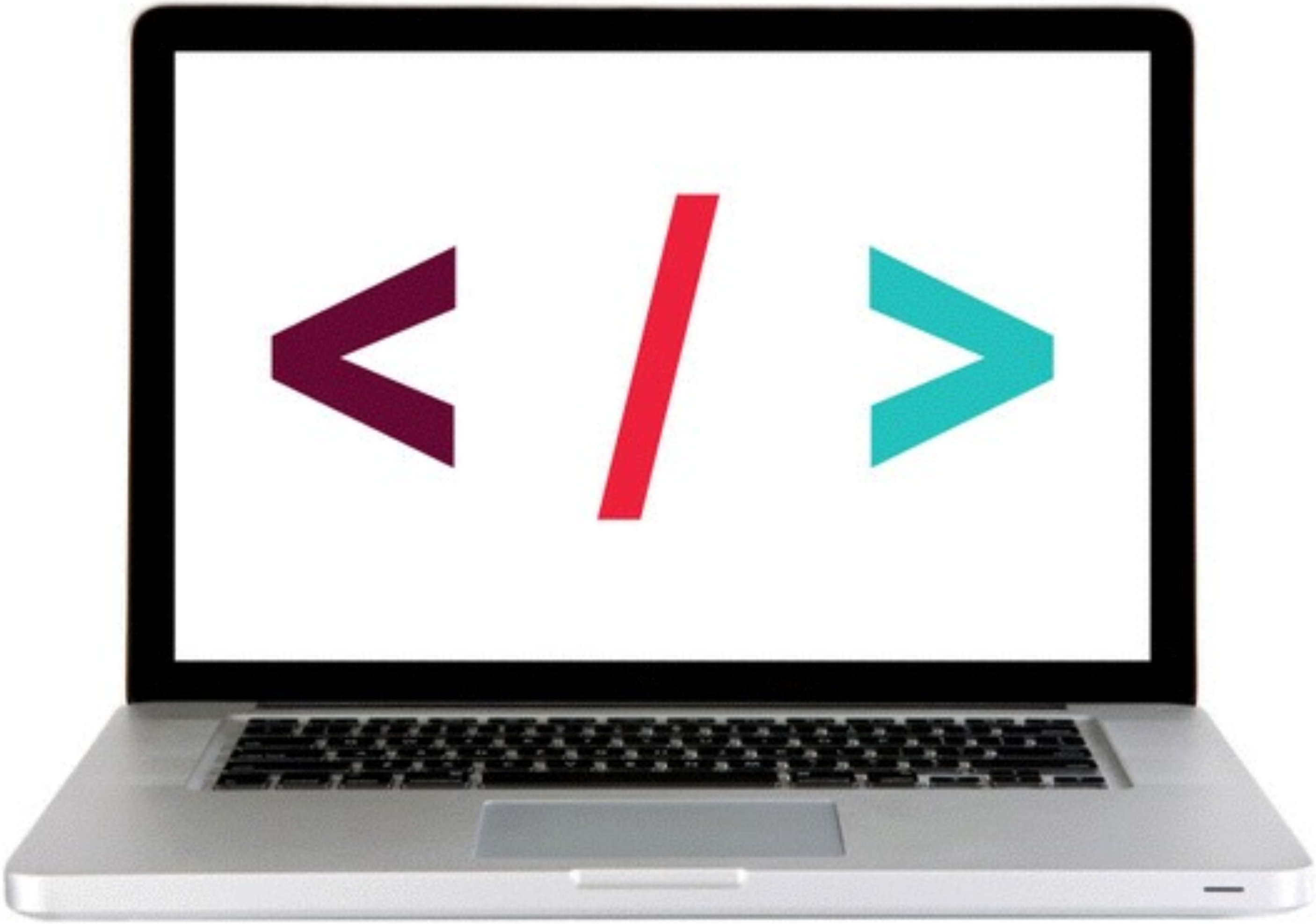
## JQUERY

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

## PURE JAVASCRIPT

- Better performance
- Faster

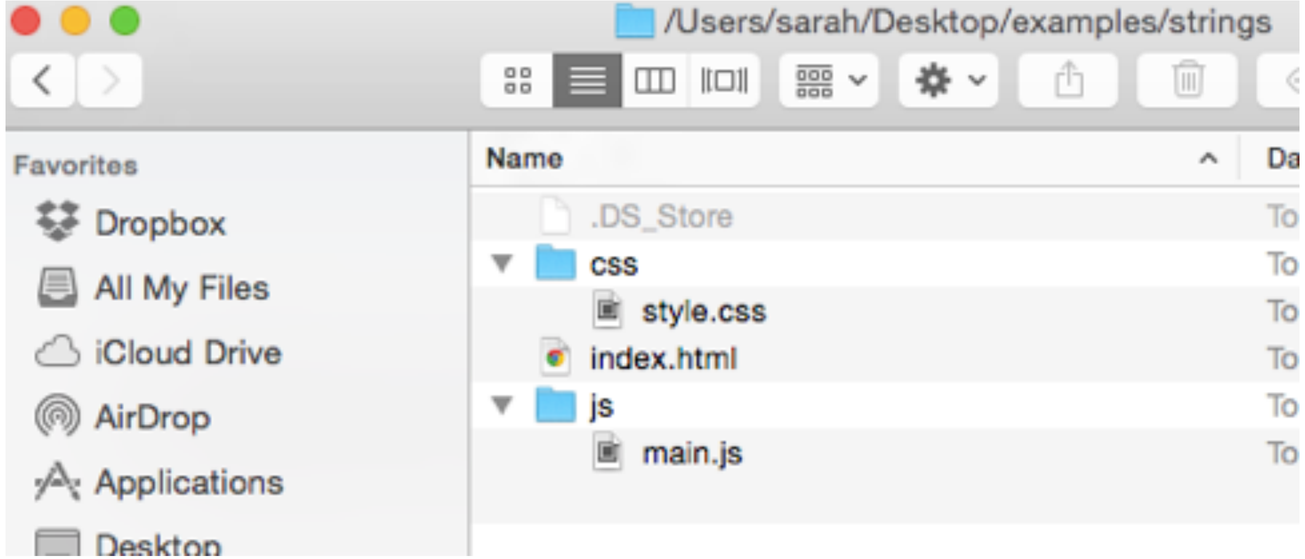
## LET'S TAKE A CLOSER LOOK – COLOR SWITCHER



# 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*.



/Users/sarah/Desktop/examples/strings

Favorites
- Dropbox
- All My Files
- iCloud Drive
- AirDrop
- Applications
- Desktop

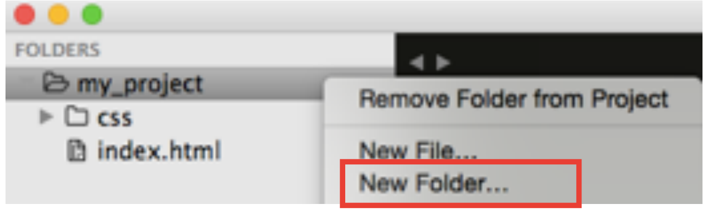
```
Name
  .DS_Store
▼ css
    style.css
  index.html
▼ js
    main.js
```

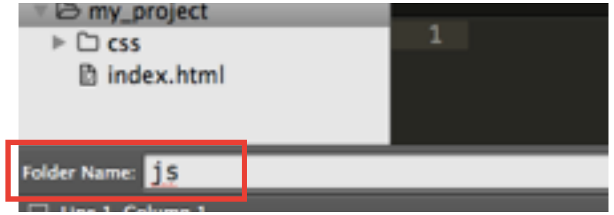
Remember - use an underscore or dash between words in folder names instead of a space. And try to avoid characters/symbols in file names (*really_cool_page.html* or *really-cool-page.html*).

## STEP 1: ADD JQUERY TO YOUR WEBSITE

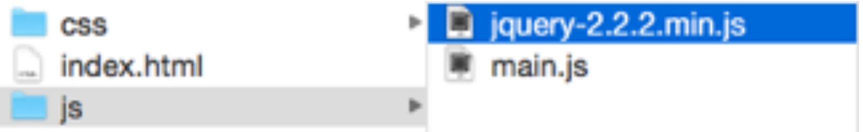
1. Download the jQuery script (version 2.x, compressed).

2. Add a js folder to your project





3. Move the jQuery file to the js folder

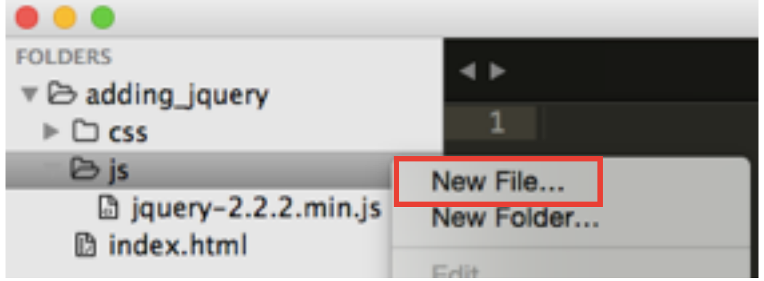


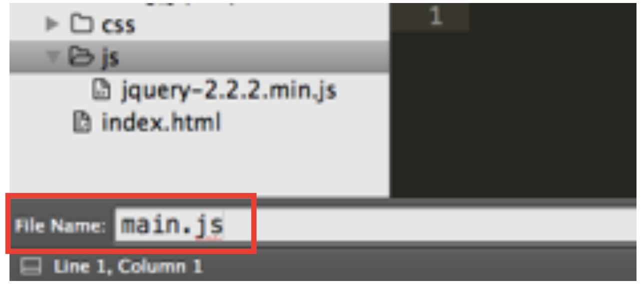
4. Use a <script> tag to include the jQuery file after your HTML content and before any other JavaScript files that use it.

```
<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)





File Name: 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>
```

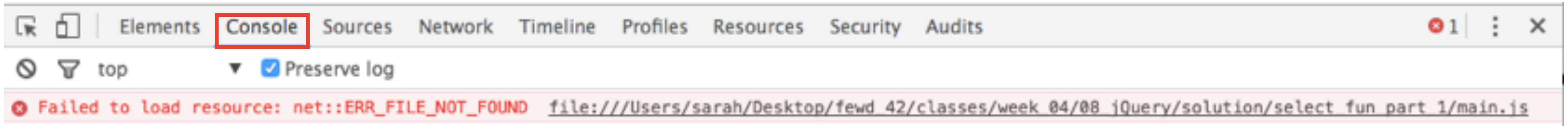
ORDER IS IMPORTANT!!!!

# MAKE SURE YOUR JS IS HOOKED UP PROPERLY

- **Method 1:** Add an alert to the top of your JS file. When you open the page in your browser, an alert will pop up if your JavaScript file is properly hooked up.

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

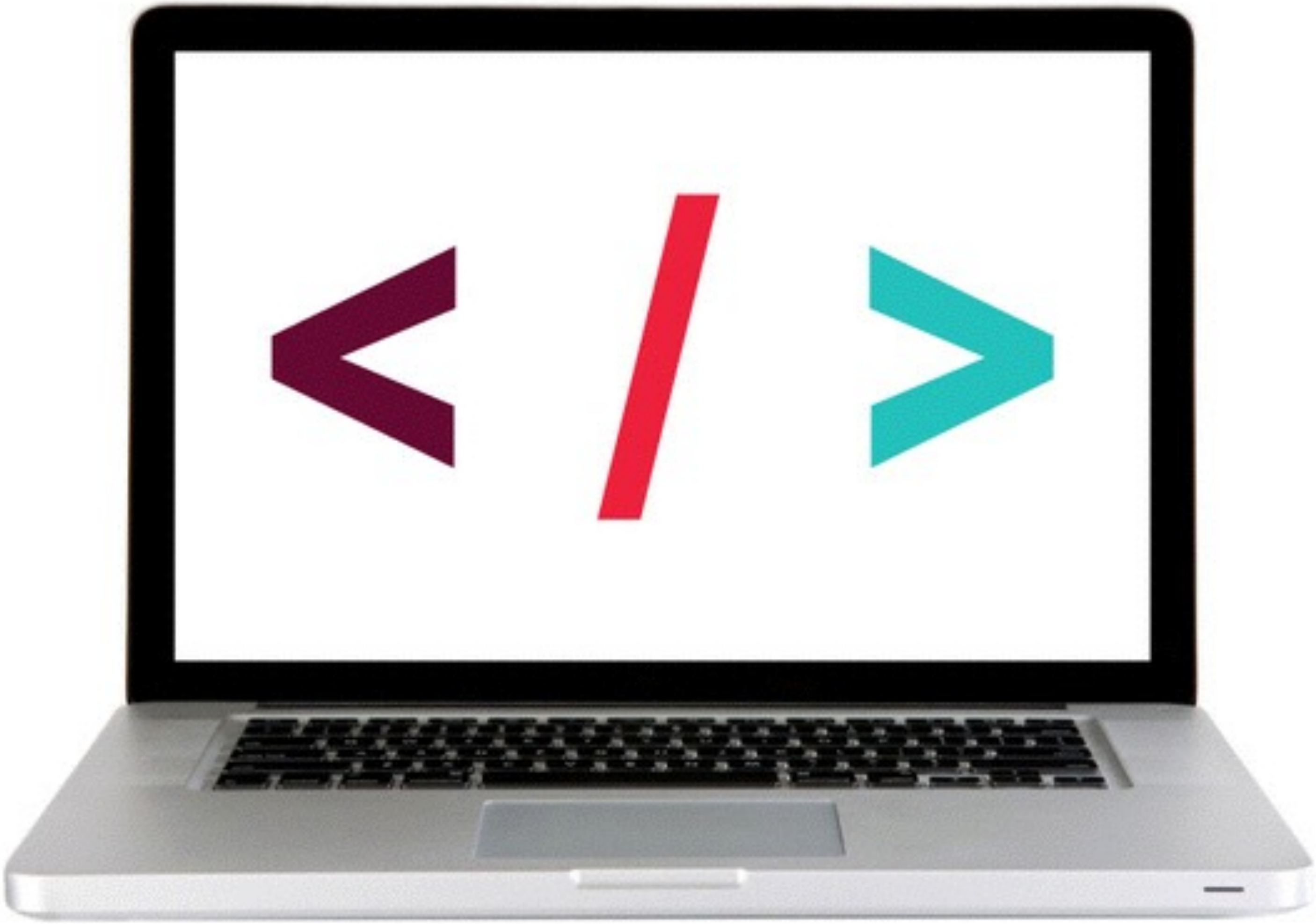
- **Method 2 (preferred):** Open the page in Chrome. Go to view > developer > developer tools. Click on the console tab and make sure there are no errors.



```
Failed to load resource: net::ERR_FILE_NOT_FOUND   file:///Users/sarah/Desktop/fewd_42/classes/week_04/08_jQuery/solution/select_fun_part_1/main.js
```

*This error means the file can't be found. Check your url in your script tag. Make sure the file exists.*

## LET'S TAKE A CLOSER LOOK



# USING JQUERY

# STATEMENTS

- Last class we chatted 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 + /*

# PART 1 — SELECT AN ELEMENT

# USING JQUERY TO MANIPULATE THE DOM

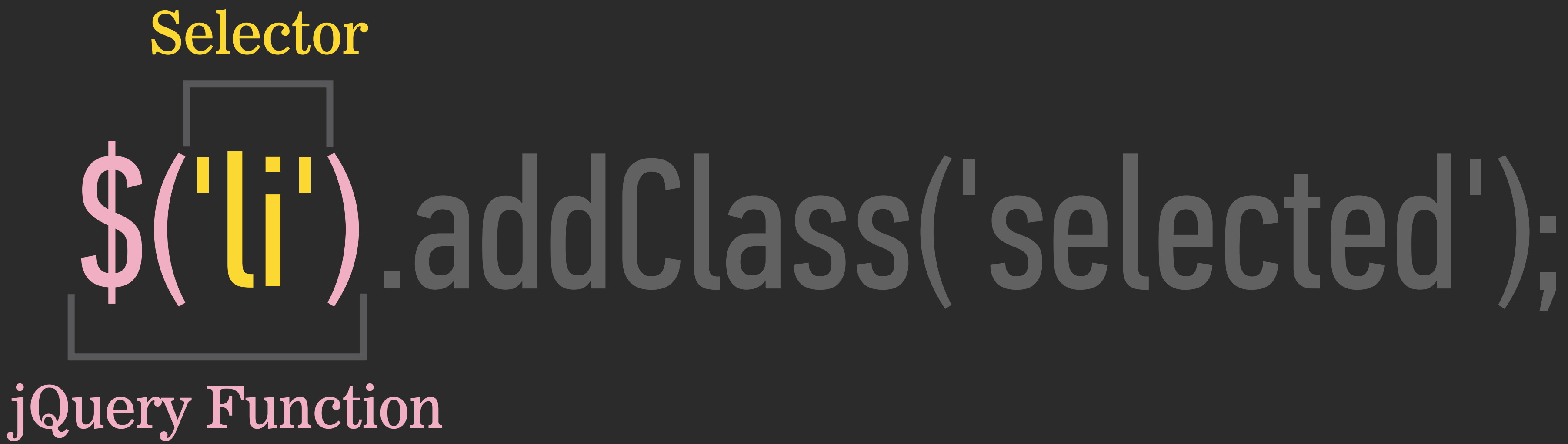
1

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

# 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!!!*

See your handout, pages 302-303 in the textbook, or the jQuery docs for list!

# PART 2 — ADD A METHOD

# USING JQUERY TO MANIPULATE THE DOM

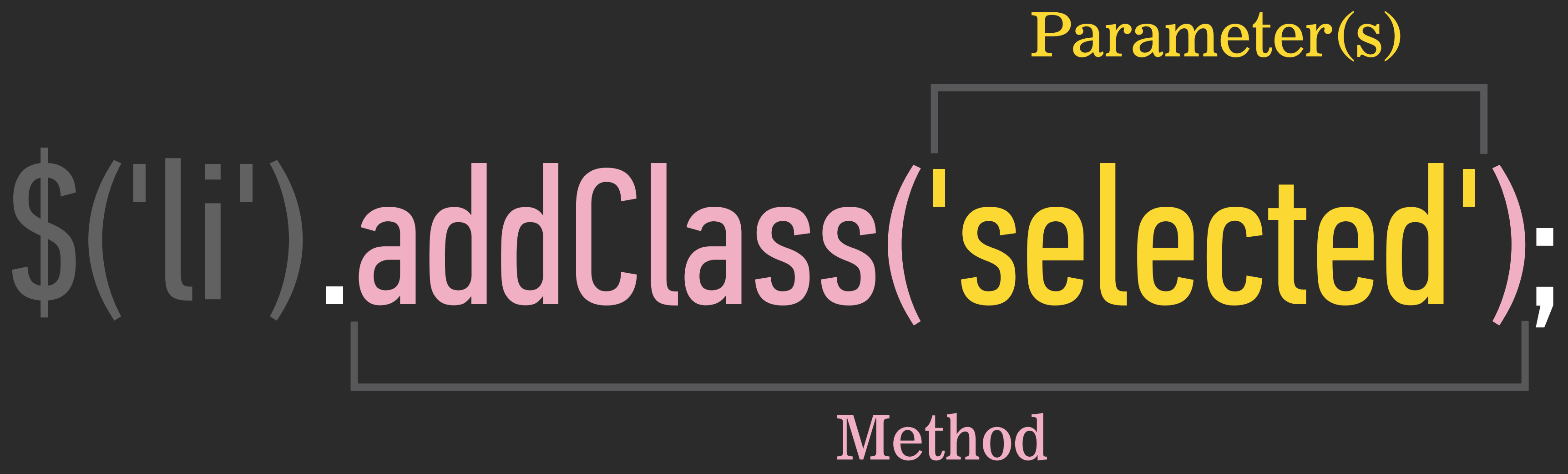
1

Select an element/elements

2

Work with those elements

# JQUERY — WORKING WITH THOSE ELEMENTS



Parameter(s)

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

Method

# 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.

# JQUERY METHODS — WORKING WITH THOSE ELEMENTS

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

FIND ELEMENTS

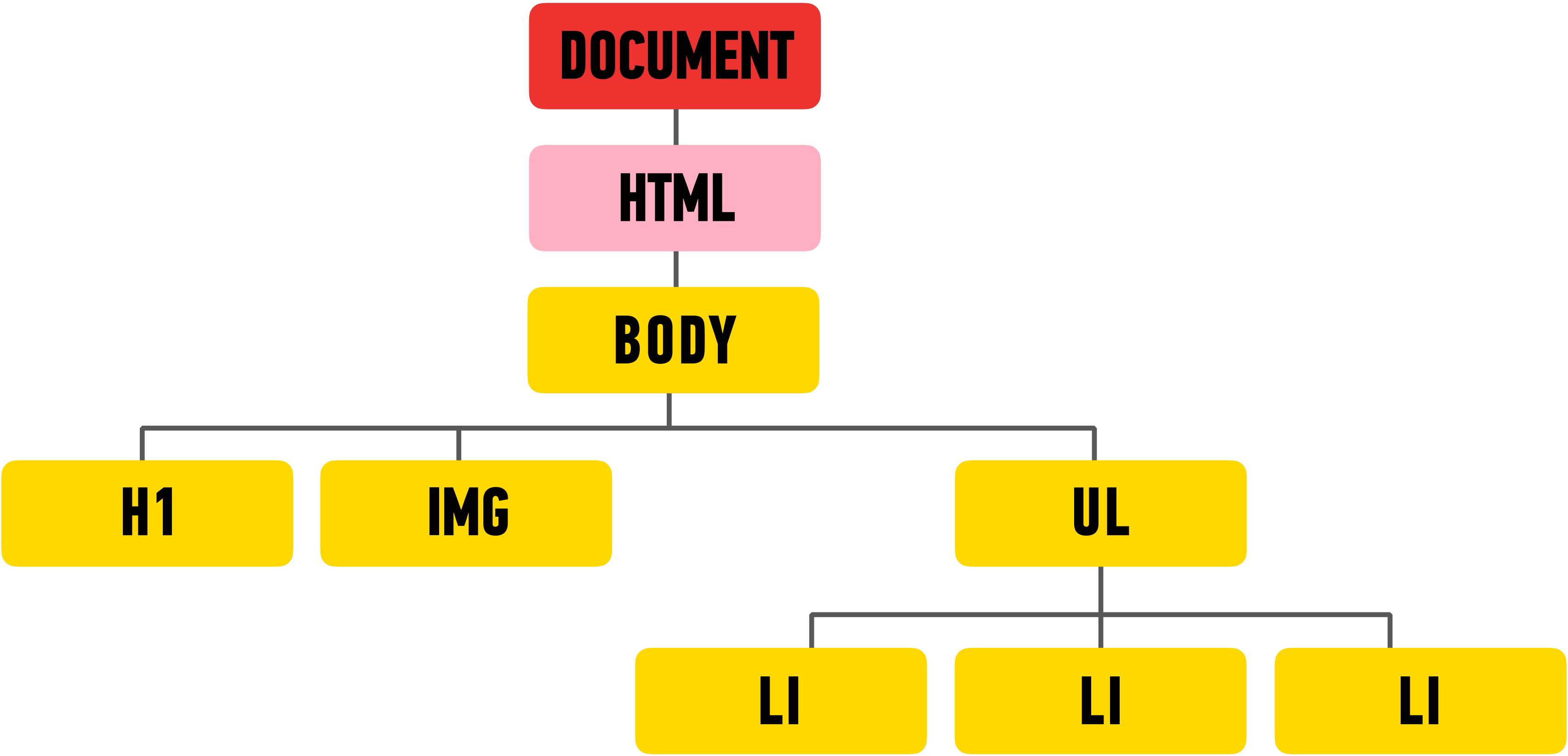
GET/SET CONTENT

ADD EFFECTS/ ANIMATION

CREATE EVENT LISTENERS

See your handout, pages 304-305 in the textbook, or the jQuery docs for list!

# TRAVERSING THE DOM?



# JQUERY METHODS — TRAVERSING THE DOM

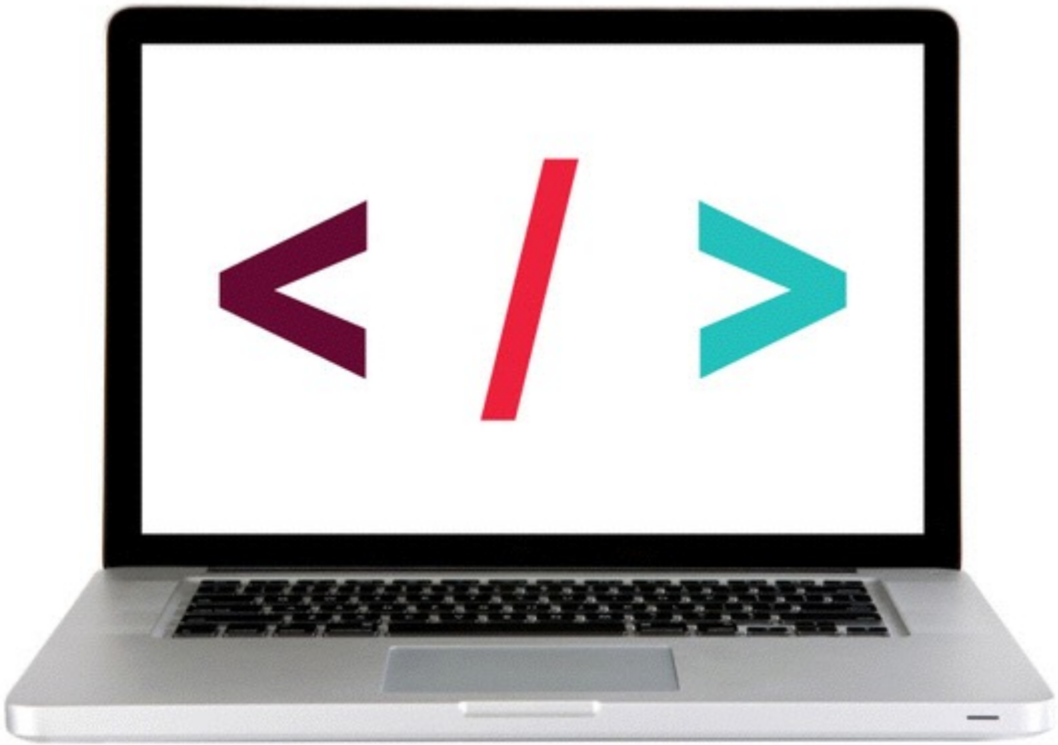
TRAVERSE THE DOM

- These methods to find/select elements to work with & traverse the DOM
- Think of these as filters, or part of the selection process.
- They must come *directly after another selection*

| METHODS | EXAMPLES |
|---|---|
| `.find()` *finds all descendants* | `$('h1').find('a');` |
| `.parent()` | `$('#box1').parent();` |
| `.siblings()` | `$('p').siblings('.important');` |
| `.children()` | `$('ul').children('li');` |

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

# LET'S TAKE A CLOSER LOOK



## JQUERY METHODS — WORKING WITH THOSE ELEMENTS

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

- FIND ELEMENTS
- 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 — GETTING/SETTING CONTENT

GET/SET CONTENT

Get/change content of elements, attributes, text nodes

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

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

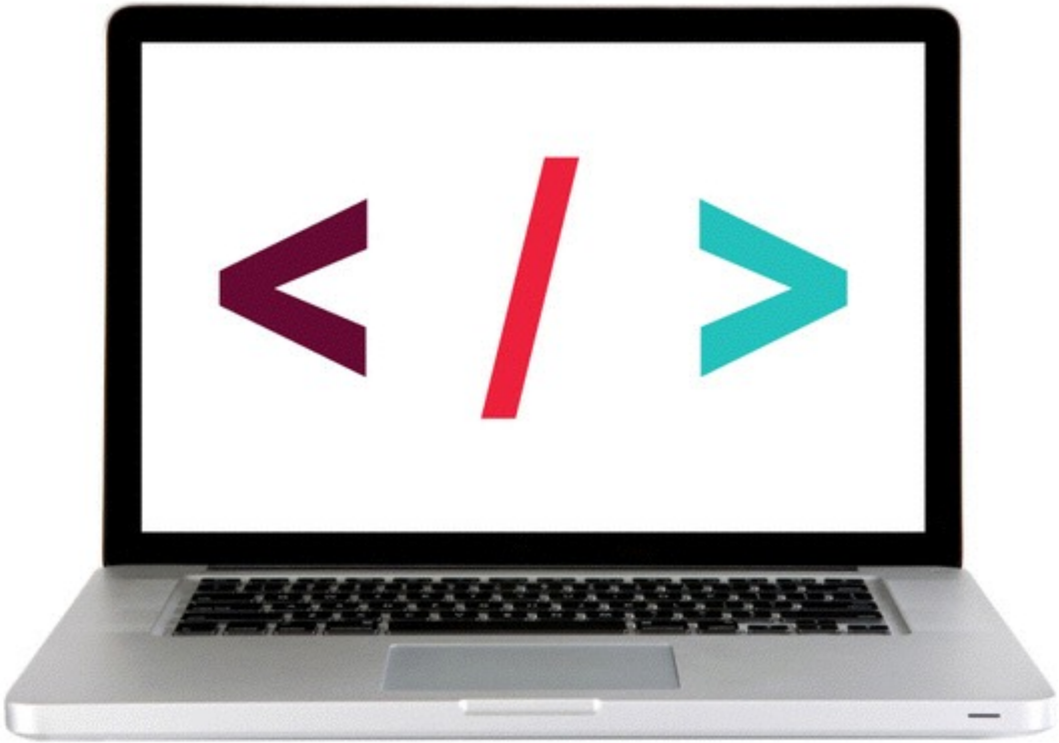
## JQUERY METHODS — GETTING/SETTING CONTENT

GET/SET CONTENT

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

NO PERIOD!!!

# LET'S TAKE A CLOSER LOOK

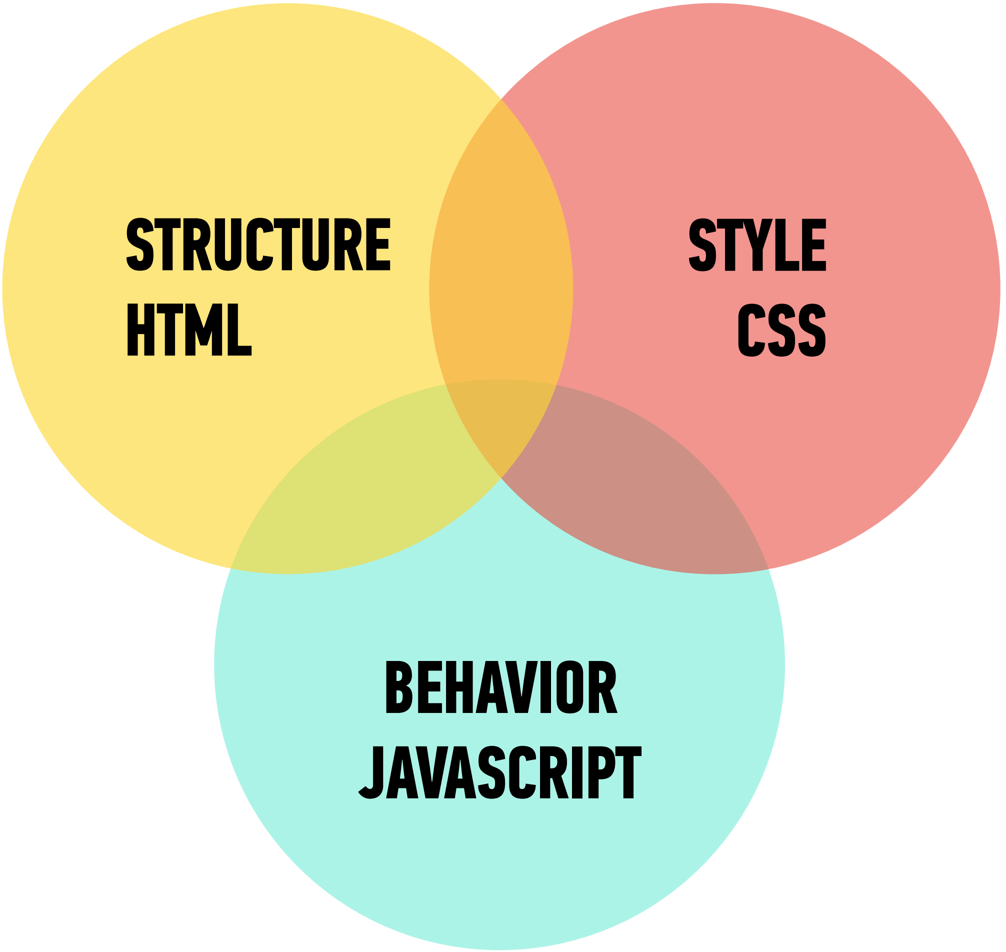


JQUERY

# SEPARATION OF CONCERNS

# THE THREE AMIGOS: STRUCTURE, STYLE, BEHAVIOR

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



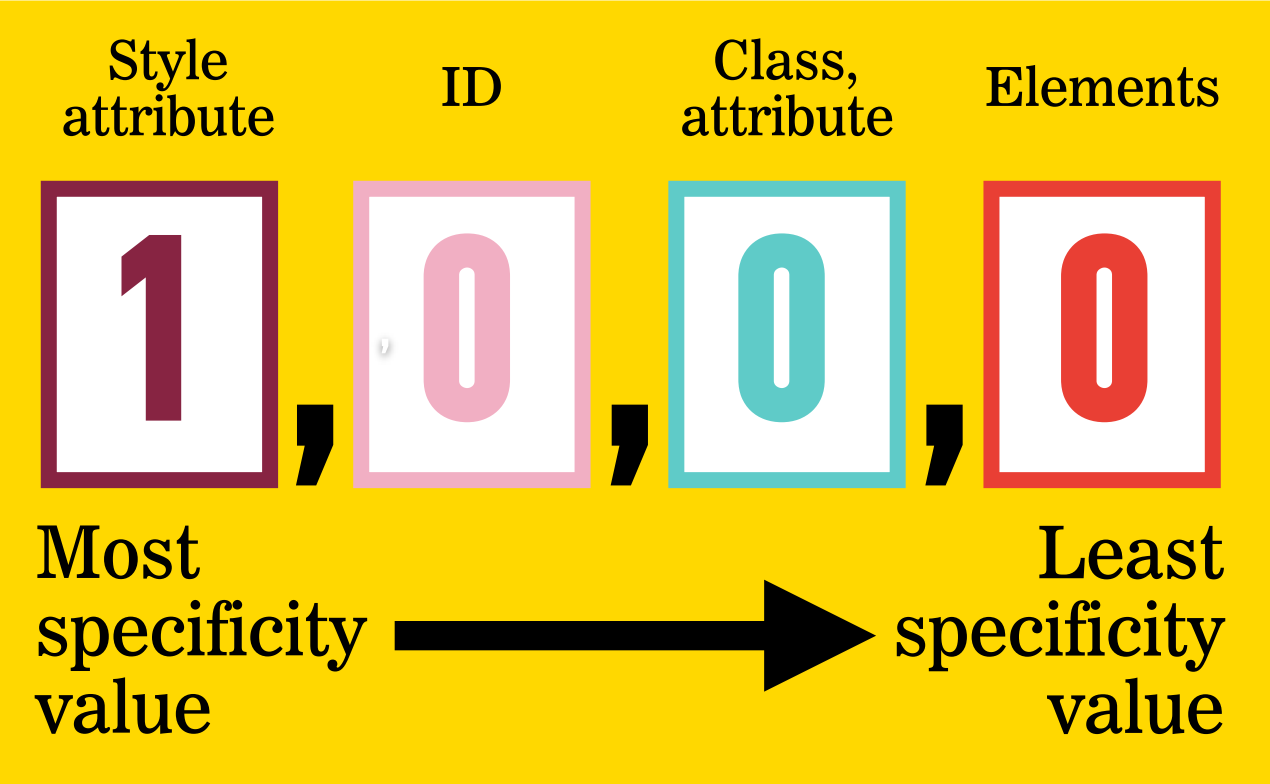
## SEPARATION OF CONCERNS

- 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 last week to follow this guideline?

# CSS CASCADING

WINNER!

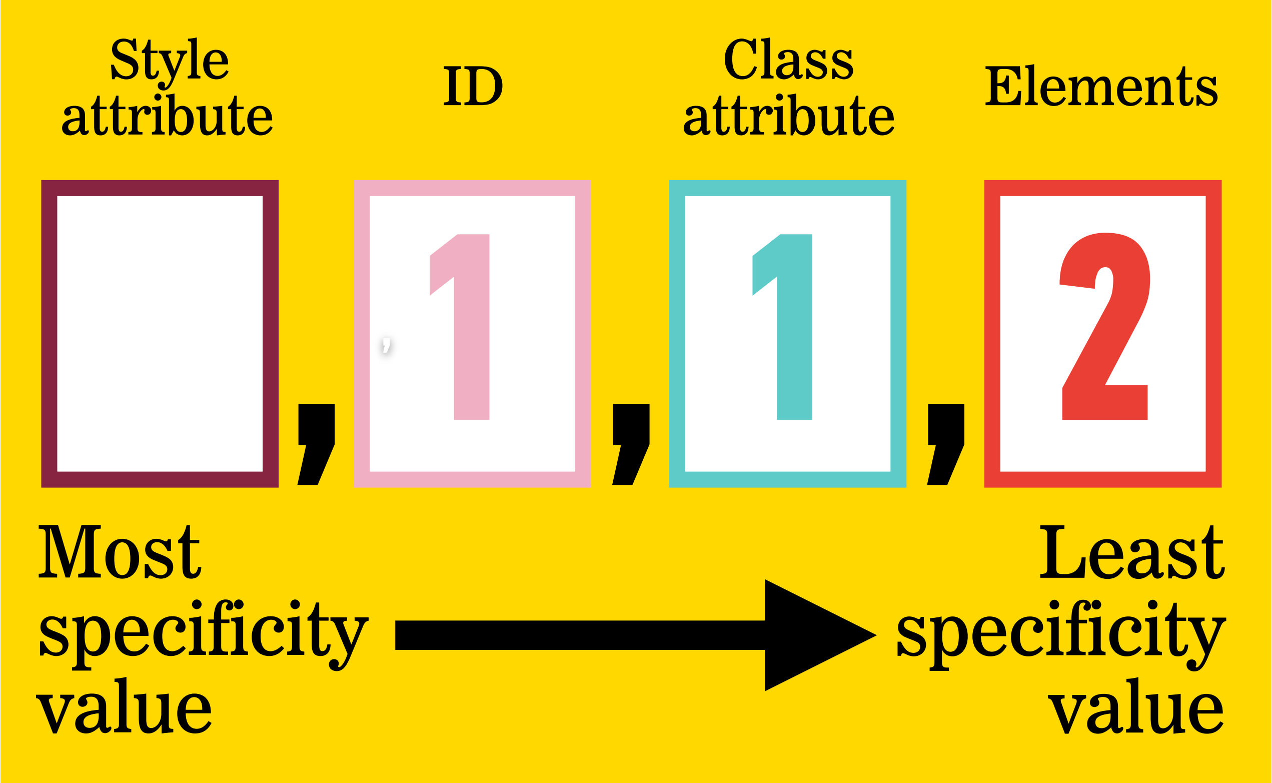
<li style="color: red"></li>



| Style attribute | ID | Class, attribute | Elements |
|---|---|---|---|
| 1, | 0, | 0, | 0 |

Most specificity value → Least specificity value

#about .first h2 + li



| Style attribute | ID | Class attribute | Elements |
|---|---|---|---|
| , | 1, | 1, | 2 |

Most specificity value → Least specificity value

# ACTIVITY

EXERCISE

### KEY OBJECTIVE

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

### TYPE OF EXERCISE

- Individual/Partner

### AS A CLASS

*Until 8:50*

Exercise is in starter_code_lesson_8 > jquery_code_along

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

# JQUERY METHODS — WORKING WITH THOSE ELEMENTS

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

FIND ELEMENTS

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 — EFFECTS/ANIMATION

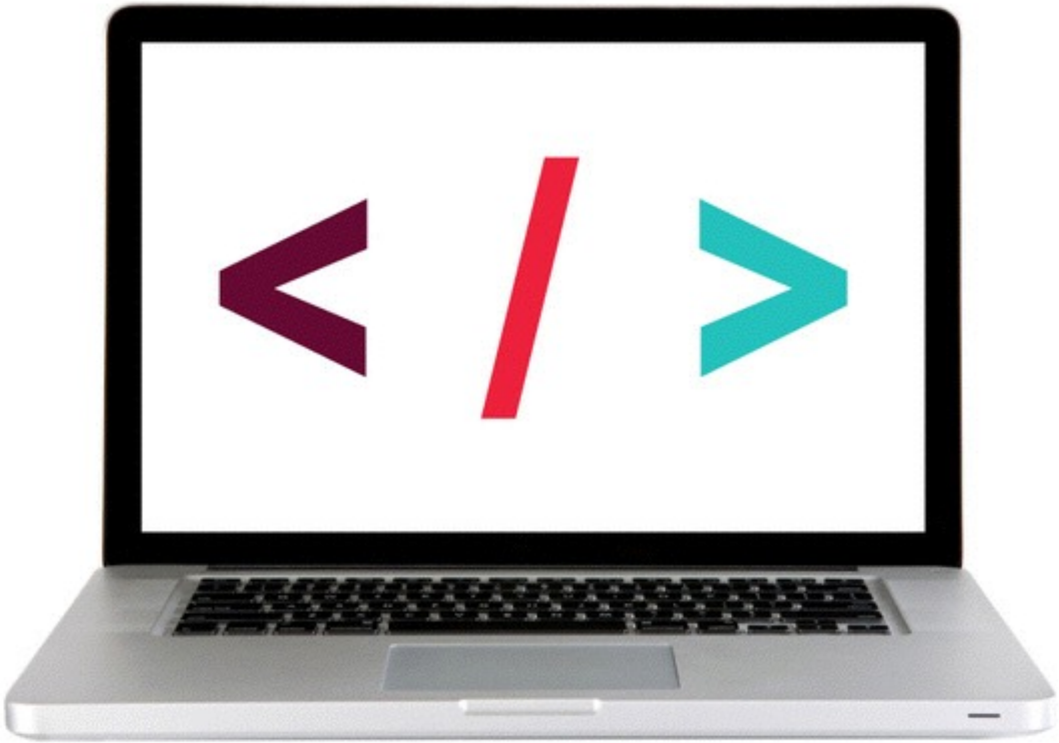
ADD EFFECTS/ ANIMATION

Add effects and animation to parts of the page

| METHODS | EXAMPLES |
|---|---|
| `.show()` | `$('h1').show();` |
| `.hide()` | `$('ul').hide();` |
| `.fadeIn()` | `$('h1').fadeIn(300);` |
| `.fadeOut()` | `$('.special').fadeOut('fast');` |
| `.slideUp()` | `$('div').slideUp();` |
| `.slideDown()` | `$('#box1').slideDown('slow');` |
| `.slideToggle()` | `$('p').slideToggle(300);` |

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:

FIND ELEMENTS

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

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

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

**Example:**

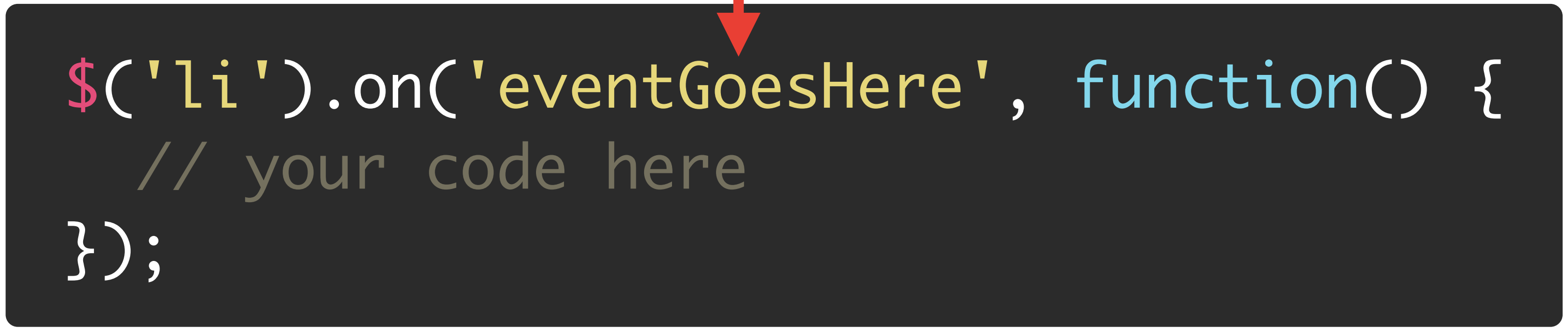
```
$('li').on('click', function() {
  // your code here
});
```

# 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
});
```

# ACTIVITY

EXERCISE

## KEY OBJECTIVE

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

## TYPE OF EXERCISE

- Individual/Partner

## AS A CLASS

*Until 8:50*

Exercise is in starter_code_lesson_8 > jquery_code_along

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

# METHOD CHAINING

# DEBUGGING

# 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

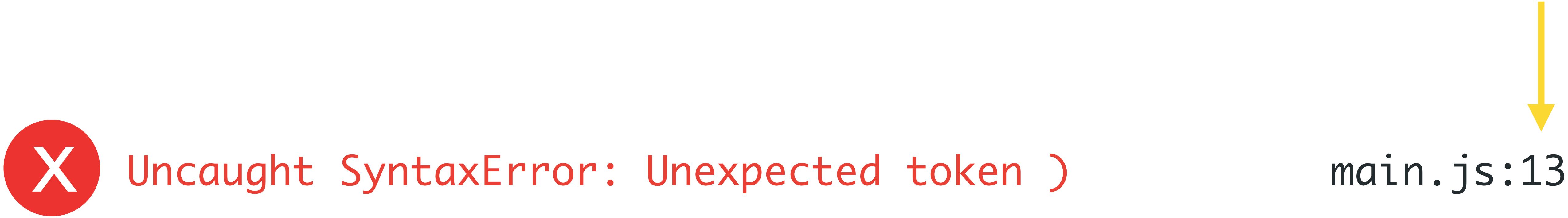
*To access debugging console:*

PC: CTRL+SHIFT+J<br>
Mac: COMMAND+OPTION+J<br>
Click the error

# DEBUGGING — LEVEL 1

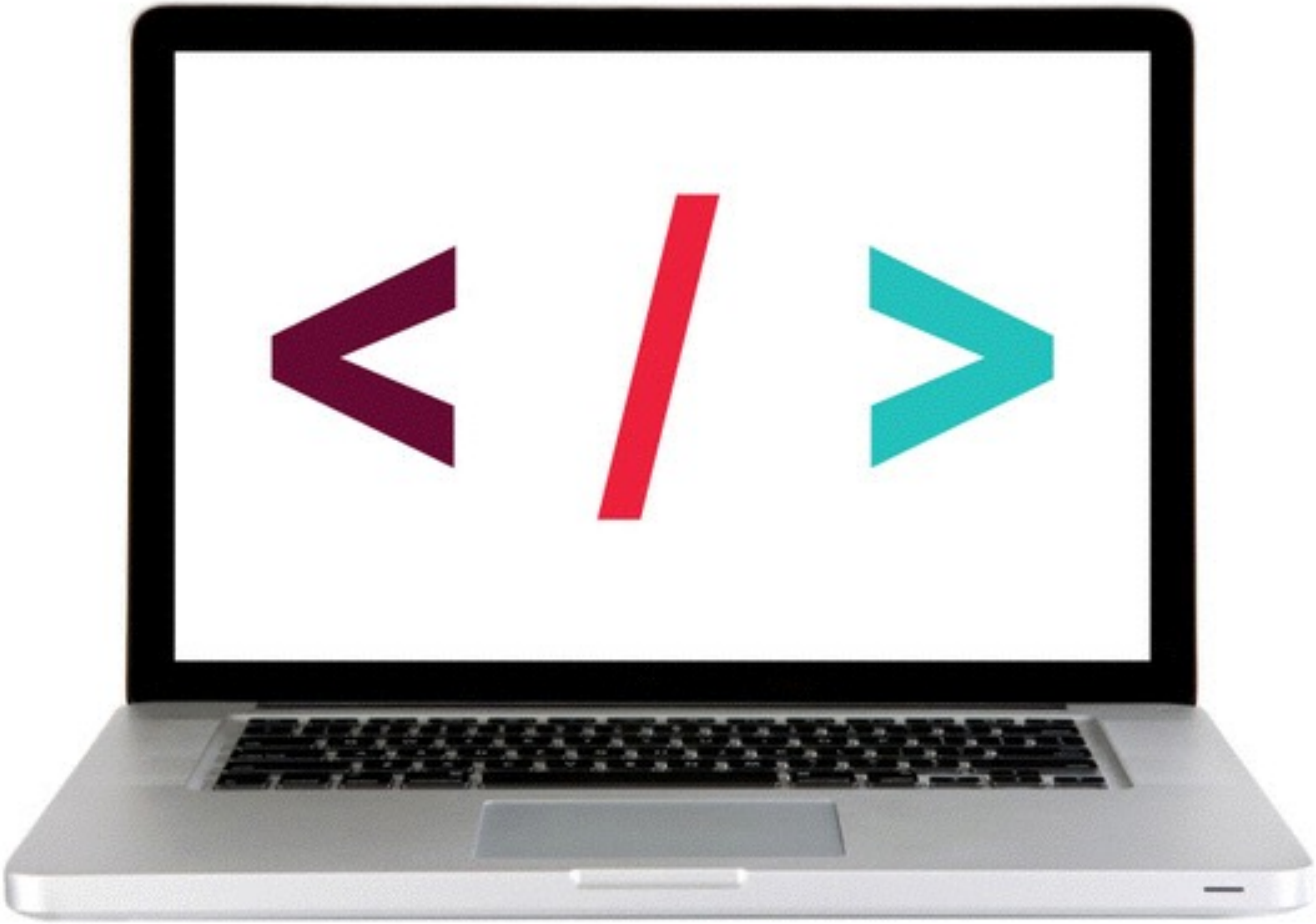
## Check for errors in console

- 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 — FIND THE BUG



# 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 instructors.

Help us help you!

1. CC both me and Jeff on any emails
2. Be descriptive about the problem.
3. Tell us what you've already done to try to figure it out.
4. Attach a .zip file of your lab.

# 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

# 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 starter_code_lesson_8 > 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 (starter_code_lesson_8 > 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.

# HOMEWORK

# HOMEWORK

Be sure to read the homework specs on the FEWD 42 Dashboard!

# EXIT TICKETS

http://goo.gl/forms/vPhCOlfESf