FOR INSTRUCTOR PURPOSES ONLY

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.

FOR INSTRUCTOR PURPOSES ONLY

INSTRUCTOR TIPS

- There’s a lot to fit into this class! Don’t worry about diving too deep—this should be an introductory lesson.
- Keep track of time, so you can get through all the content.
- Be aware of differing skill levels and try to differentiate as needed.

FOR INSTRUCTOR PURPOSES ONLY

MATERIALS

- Laptop
- Google Chrome
- Sublime Text or Atom

FOR INSTRUCTOR PURPOSES ONLY

PRE-WORK

- DASH Project 1
- Install either Sublime Text 3 or Atom on to computer (choose one before telling students which one to install).
- Write learning objectives on board.
- Make sure wifi network and password (GA Guest, yellowpencil) is written on board, since students will need to be online to use codepen.io.

GENERAL ASSEMBLY

JAVASCRIPT 101

Insert Instructor Name

Title, Company

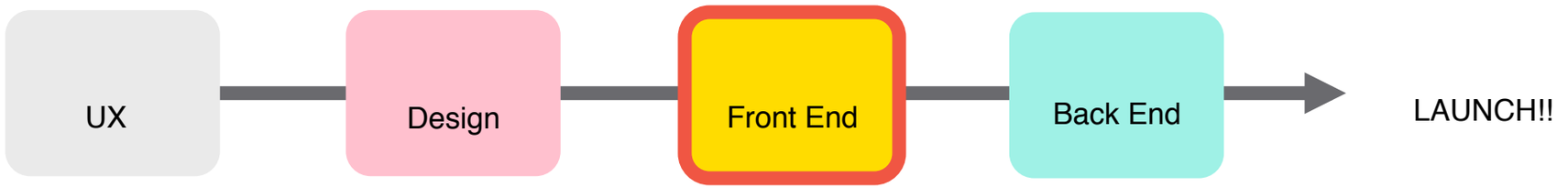
LEARNING OBJECTIVES

- 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.

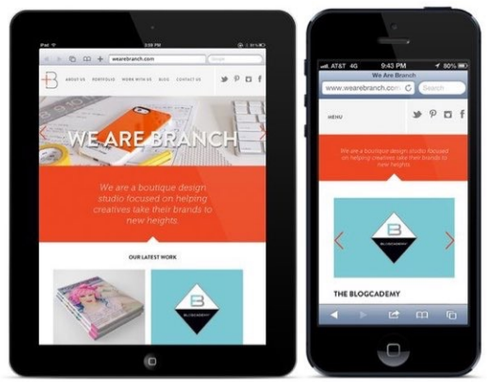
WHERE DOES FRONT END WEB DEVELOPMENT FIT IN?

WEB PRODUCTION WORKFLOW

WEB PRODUCTION WORKFLOW



- Take the 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



WHAT DOES FRONT END EVEN MEAN?

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

FRONT END IS MADE UP OF…

HTML

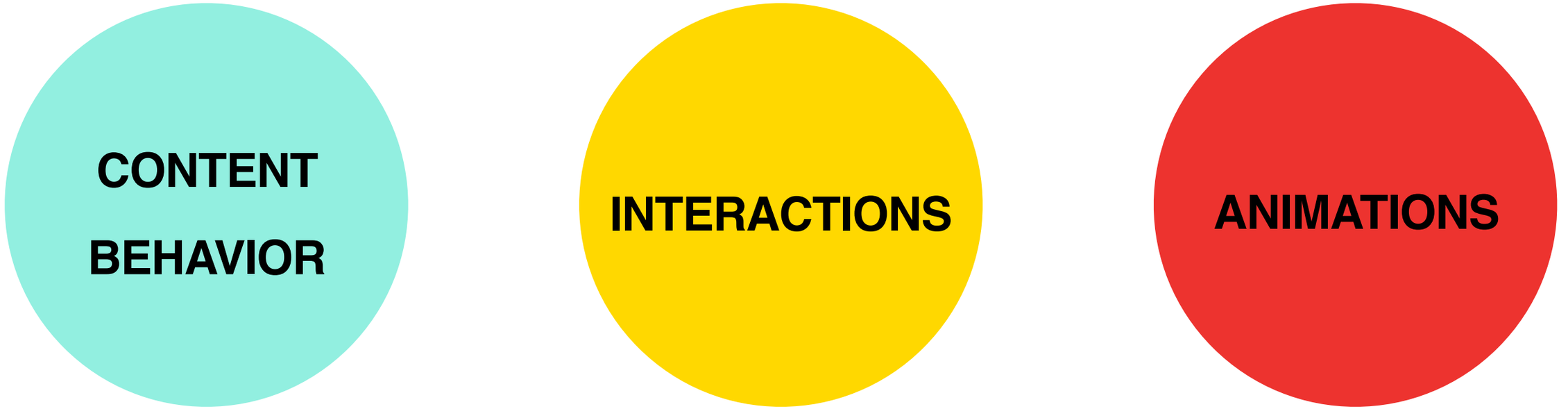
CSS **JavaScript**

IN SUMMARY…

- 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.

WHAT IS JAVASCRIPT?

JAVASCRIPT IS RESPONSIBLE FOR...



WITH JAVASCRIPT YOU CAN…

1 Access Content

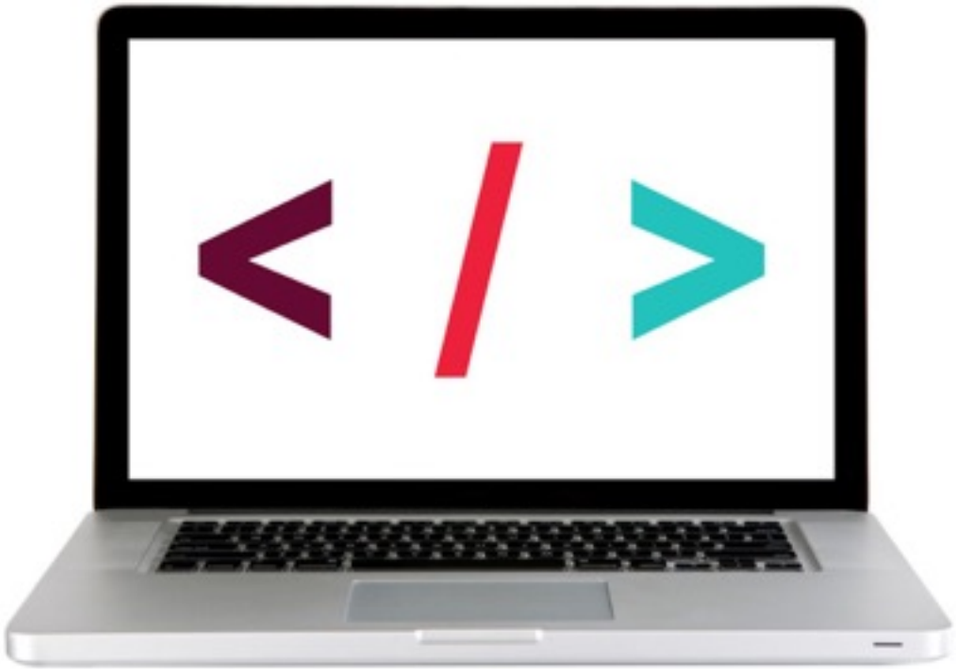
2 Modify Content

3 Program Rules

4 React to Events

MANIPULATE THE DOM

LET'S TAKE A LOOK



https://lyft.com/

ACTIVITY: PRACTICE READING JS

DIRECTIONS

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

INTRO TO PSEUDO CODE

PSEUDO CODE IS…

- 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.

THE IMPORTANCE OF PLANNING

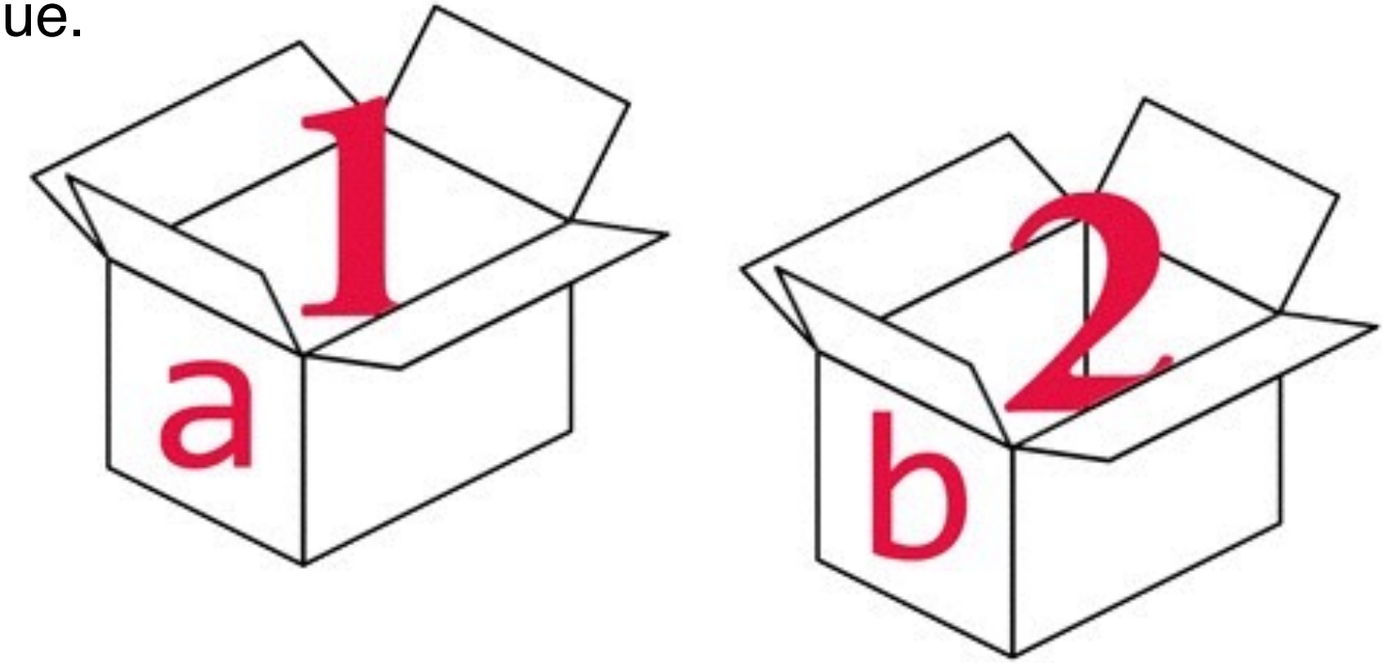
PSEUDO CODE — THERMOSTAT

Goal: Write pseudo code for an application that would monitor the room temperature and adjust it so the room remains at a certain temperature.

VARIABLES

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 a value.
- The value can change.




```
var age = 29;
```

VARIABLES

- **Reassigning a variable** means you are changing the value you set initially
- How is this done?:

```
var age = 29;
age = 30;
```

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
```

Can have one of two values:

- True
- False

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
ADD VALUE TO VARIABLE	var num = 8	+=	num += 6	14
SUBTRACT VALUE FROM VARIABLE	var num = 8	-=	num -= 6	2

ACTIVITY: VARIABLES

DIRECTIONS

1. Open up the project called “Variables” in Sublime Text. I will walk you through how to do this
2. Go to the file called main.js
3. We will do Part 1 together
4. You will do Part 2 on your own

JAVASCRIPT IN ACTION

METHODS AND PROPERTIES OF STRINGS

MAKE STRING LOWERCASE:

```
var str = "Hello World";
var res = str.toLowerCase();
// the result of res will be:
// hello world
```

LENGTH OF A STRING (PROPERTY):

```
var str = "Hello World";
var n = str.length;
// the result of n will be 11
```

MAKE STRING UPPERCASE:

```
var str = "Hello World";
var res = str.toUpperCase();
// the result of res will be:
// HELLO WORLD
```

**Find a whole list of methods and properties for strings here

STRING CONCATENATION

- To take two strings and stick them together, use the + operator.
- This is called string concatenation.

```
var book = "Happy";
var summary = "Best book ever.";
var review = book + ": " + summary;
// Result will be: Happy: Best book ever.
```

HOW IS jQUERY DIFFERENT FROM JAVASCRIPT?

WHAT IS JQUERY?

- jQuery is a JavaScript file you include in your pages.
- Makes it faster and easier to write cross-browser JavaScript.
- Allows us to find elements using CSS-style selectors and then do something to them using jQuery methods.

JQUERY VS. JAVASCRIPT

JS:

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

JQUERY:

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

JQUERY VS. JAVASCRIPT

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

HOW DOES jQUERY WORK?

USING JQUERY TO MANIPULATE THE DOM

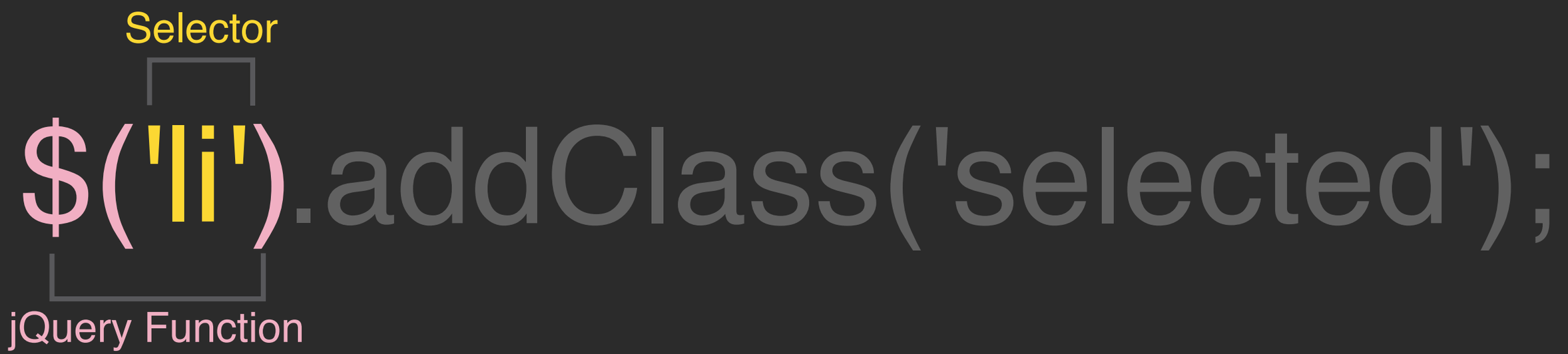
1

Select an element/elements

2

Work with those elements

JQUERY — SELECTING ELEMENTS



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!!!

USING JQUERY TO MANIPULATE THE DOM

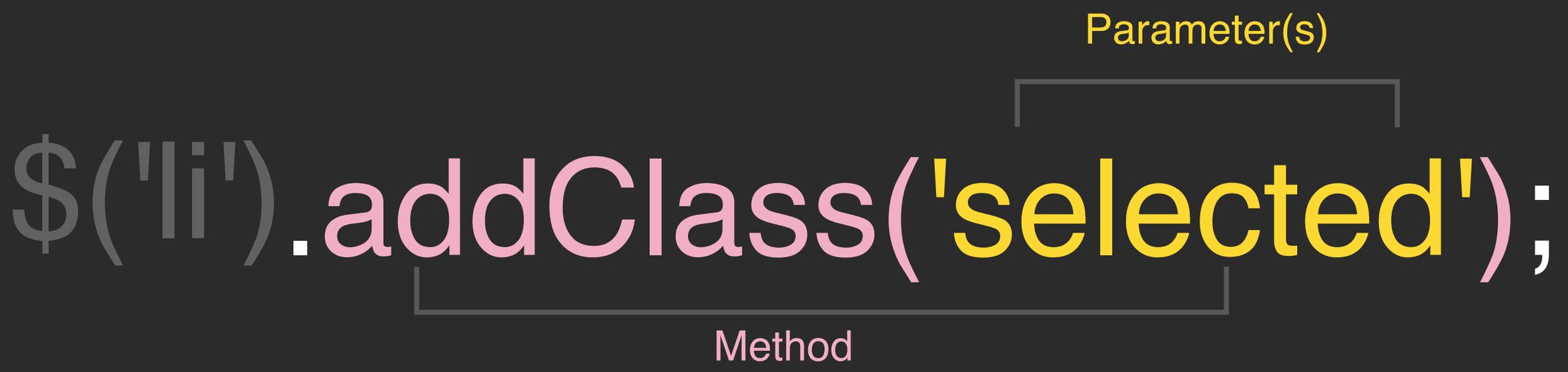
1

Select an element/elements

2

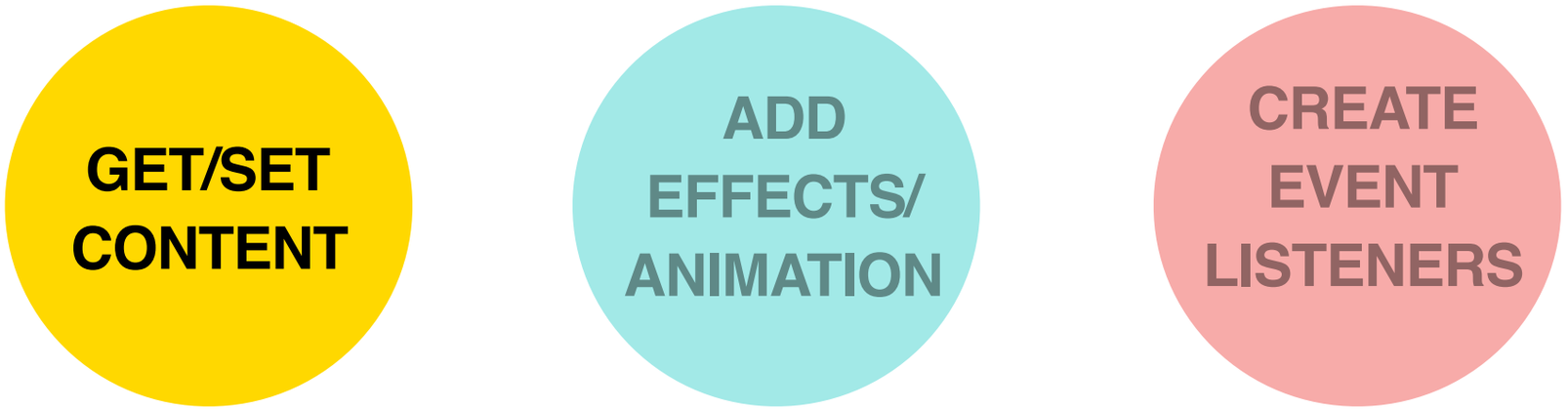
Work with those elements

JQUERY — WORKING WITH THOSE ELEMENTS



JQUERY METHODS — WORKING WITH THOSE ELEMENTS

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



Refer to the jQuery docs for list

JQUERY METHODS — GETTING/SETTING 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');
```

ACTIVITY

KEY OBJECTIVE

- Utilize jQuery to access and manipulate DOM elements.

TYPE OF EXERCISE

- Individual/Partner

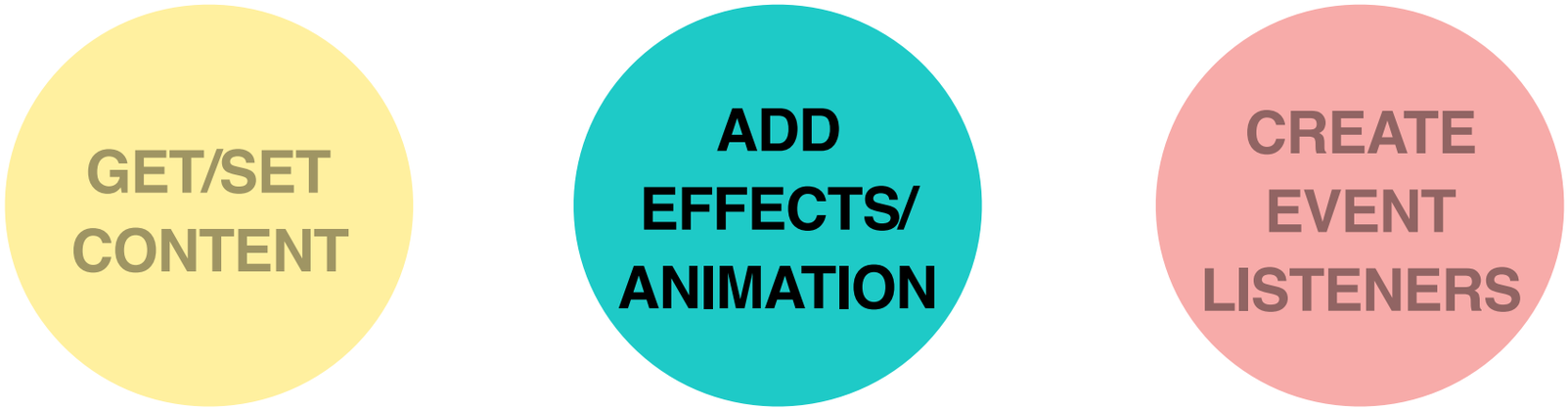
AS A CLASS

Exercise is in starter_code > jquery_code_along

1. Follow the instructions under part 1 in main.js

JQUERY METHODS — WORKING WITH THOSE ELEMENTS

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



Refer to the jQuery docs for list

JQUERY METHODS — EFFECTS/ANIMATION

Add effects and animation to parts of the page

Some methods available to us:

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

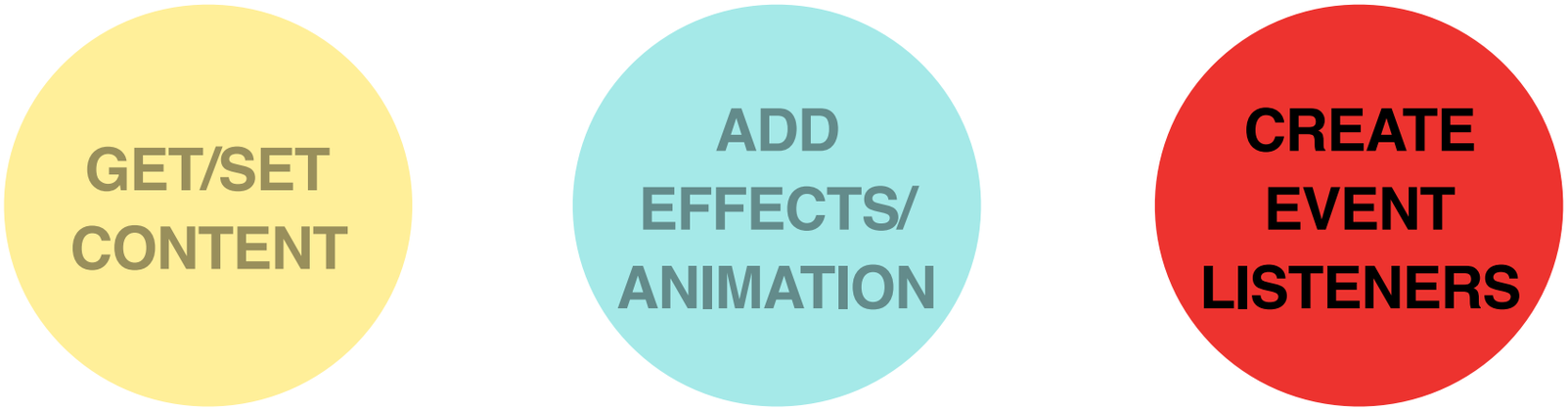
What goes in the parenthesis?
An animation speed

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:



Refer to the jQuery docs for list

JQUERY METHODS — EVENTS!

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!

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

KEY OBJECTIVE

- Utilize jQuery to access and manipulate DOM elements.

TYPE OF EXERCISE

- Individual/Partner

AS A CLASS

Exercise is in starter_code > jquery_code_along

1. Follow the instructions under Part 2 in main.js

IF

CONDITIONALS

CONDITIONAL LOGIC

If something is true, do one thing. If it is not, do something else. This type of logic or statement is a condition.

In JavaScript (and coding in general) you'll need to make comparisons all the time:

- Is a user logged in?
- Has the user chosen three or more colors?
- Is the password correct?
- Does a user have enough money in their bank account?
- etc.

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

or


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

IF STATEMENTS

```
if (age > 65) {
    $('h1').html("Senior Discount Applied");
}
```

IF STATEMENTS

```
if (age > 65) {
    $('h1').html("Senior Discount Applied");

} else {
    $('h1').html("Sorry, you do not qualify for a discount.");
}
```

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
}
```

IF STATEMENTS

```
if (age > 65) {
    $('h1').html("Senior Discount Applied");

} else if (age < 18) {
    $('h1').html("Student Discount Applied");

} else {
    $('h1').html("Sorry, you don't qualify for a discount");
}
```

JAVASCRIPT — LOGICAL OPERATORS

&& and

|| or

! not

MULTIPLE CONDITIONS

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

WRAP-UP

SO, WHO NEEDS TO KNOW ABOUT THIS STUFF?

WRAP UP

Business Managers

Product Managers

Designers

Marketers

Programmers

HOW TO LEARN THIS AT GA?

PART-TIME:

JS/FEWD

ONLINE:

HCD

FULL-TIME:

WDI / WDIR

REFERENCE PAGES:

- Mozilla Developer Network
- W3Schools
- CSS Tricks
- Web Field Manual
- Free IT Books
- JSDB.io
- Site Point
- Codrops

STAYING CURRENT

- Smashing Magazine
- Sidebar.io
- Codrops
- Medium
- A List Apart
- Web Designer Weekly
- Creative Bloq
- CSS Tricks
- Codepen.io

SITE INSPIRATION

- Awwwards
- Site Inspire
- Mediaqueri.es
- One Page Love
- Little Big Details
- The Best Designs
- CSS Design Awards
- Web Design File
- Behance
- Dribbble
- Responsive Patterns

A COUPLE OF LAST THINGS

You are going to get lots of errors…

Google first, ask questions later (but don't be afraid to ask)
Stack Overflow is great for this stuff!

Get immersed! Coding is hard if you don't get really involved.

ACTIVITY: AFTER CLASS YOU CAN…

DIRECTIONS

1. Go to dash.ga.co
2. The last two projects cover what we went over tonight

Q & A

THANKS!

NAME

- Optional Information:
- Email?
- Website?
- Twitter?