FILL OUT MIDCOURSE EVALUATION

FINAL PROJECTS

MAKE NEW REPOSITORY IN YOUR GITHUB TITLED FEWD FINAL PROJECT

UPLOAD BOILERPLATE TO THE REPOSITORY USING GIT WORKFLOW

HAVE CONTENT AND HTML SKELETON IN BOILERPLATE CODE READY THIS SUNDAY

AGENDA

- Review
- More javascript methods
- Objects
- Lab — Image Carousel

LEARNING OBJECTIVES

- Describe the concept of "this" as it applies within jQuery functions.
- Know how to make javascript objects and then to use them

REVIEW

JQUERY — SELECTING ELEMENTS

jQuery Function:

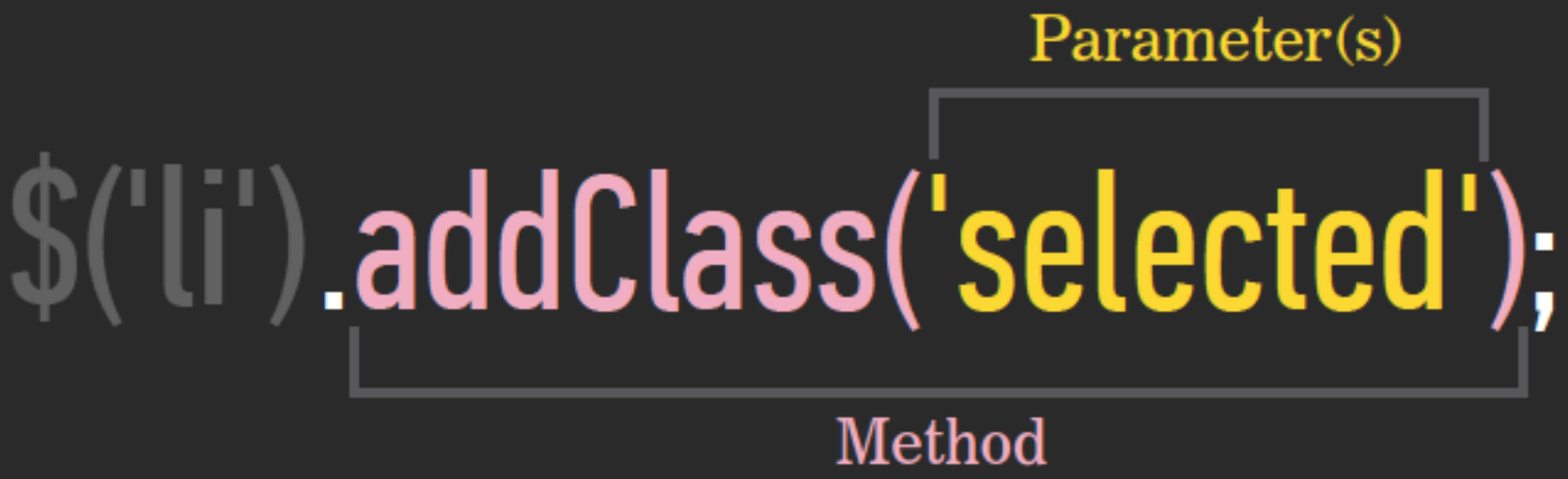
- Lets us find one or more elements in the page
- Creates a jQuery object which holds references to those elements

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')

JQUERY — WORKING WITH THOSE ELEMENTS



JQUERY METHODS — GETTING/SETTING 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');

JQUERY METHODS — EFFECTS/ANIMATION

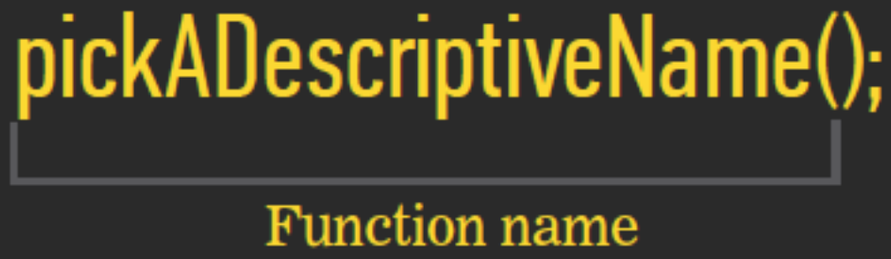
Add effects and animation to parts of the page

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);

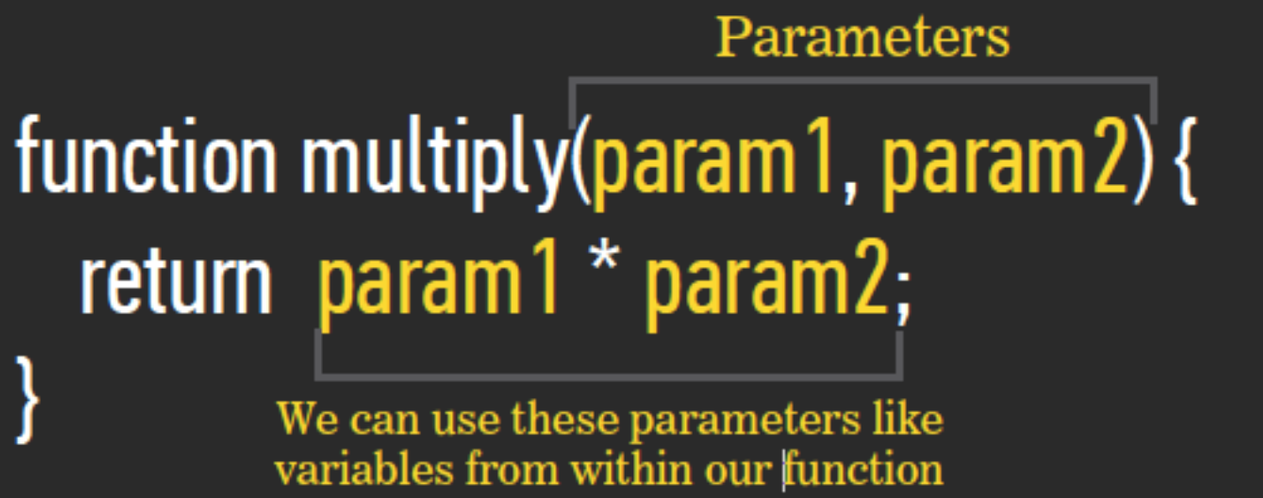
SYNTAX — DECLARING A FUNCTION

SYNTAX — CALLING A FUNCTION

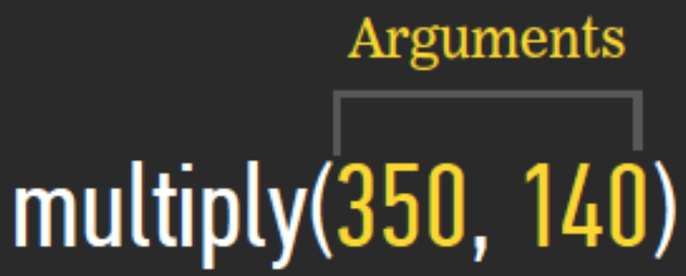
- To run the code in a function, we 'call' the function by using the function name followed by parenthesis.



SYNTAX — DECLARING A FUNCTION (WITH PARAMETERS)



SYNTAX — CALLING A FUNCTION (WITH ARGUMENTS)



STORING LISTS OF VALUES

- An array can be used to **store a list of values in a single variable**
- Holds an ordered collection of values
- Can hold numbers, strings, even other arrays!
- Good for things like a grocery list, a list of states, or any other list

DECLARING ARRAYS

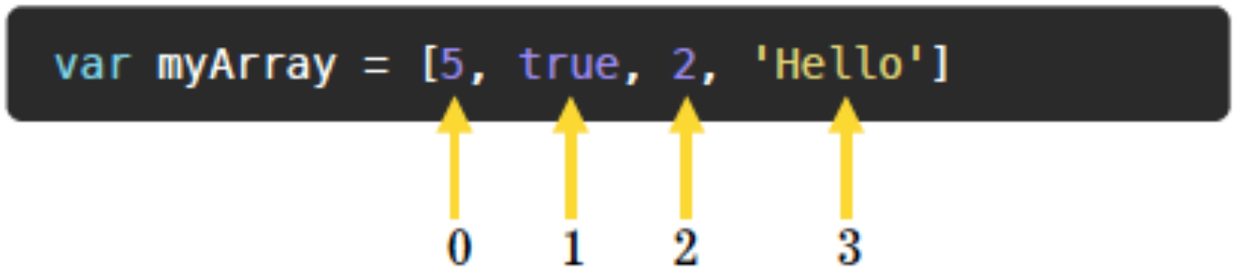
```
var descriptiveNameHere = [item1, item2, item3];
```

ARRAYS - INDEXING

- Each item in an array has an **index**, by which you can access that item.
- The first item has an index of **0**, the second item 1, the third item 2, etc.

ARRAYS - ACCESSING ITEMS BY INDEX

- Each item in an array has an **index**, by which you can access that item.
- The first item has an index of **0**, the second item 1, the third item 2, etc.



ARRAYS — ACCESSING ITEMS IN AN ARRAY

Accessing items in array:

`myArray[1]` → true

`myArray[2]` → 2

`myArray[0]` → 5

`myArray[3]` → 'Hello'

```
var myArray = [5, true, 2, 'Hello']
```

ARRAYS - ADDING A VALUE/REPLACING A VALUE

INSERTING A NEW VALUE

- We can insert new values into any space in the array using the positions index.

```
myArray[1] = 'Hello';
```

UPDATING VALUES

- If there's already an item at that position, it will be replaced with the new value.

```
var myArr = [65, 'hello', true];
myArr[1] = 'goodbye';
// myArr[1] now holds 'goodbye' instead of 'hello'
```

ARRAYS - LENGTH

- We can use the .length property to find out how many items are in an array

```
var shapes = ['circle', 'triangle', 'square'];
```

```
shapes.length;
```

=> 3

- Accessing the last element in an array:

```
console.log(shapes[shapes.length-1]);
```

=> Prints 'square'

KEEPING TRACK OF CLASSES AND STATES

HAS CLASS

jQuery's .hasClass() method is an easy way to tell whether or not an item is in a particular state.

```
$('h2').on('click', function () {
  var isSelected = $('h2').hasClass('selected');
});
```

EXERCISE — BLACKOUT

KEY OBJECTIVE

- Practice combining conditionals with jQuery to create a simple interaction.

TYPE OF EXERCISE

- Individual/Paired

EXECUTION

5 min	1. Together: write pseudo code for the light/dark switcher.
8 min	2. In pairs, write code to complete the light/dark switcher

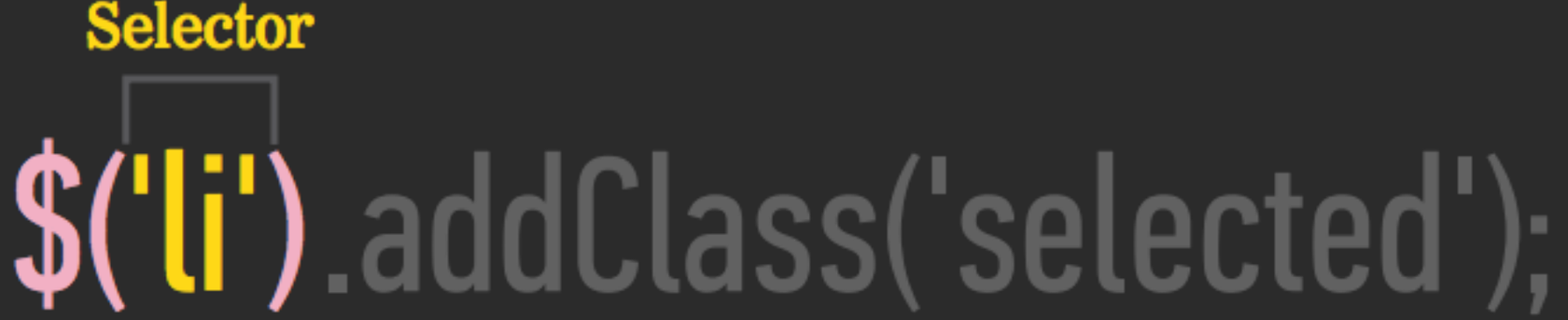
‘THIS’ KEYWORD

THE KEYWORD 'THIS'

this refers to whatever you *selected* with jQuery

Notice — no quotes around this!

JQUERY — SELECTING ELEMENTS



DECLARING ARRAYS

EXERCISE

KEY OBJECTIVE

- Practice applying the this keyword

TYPE OF EXERCISE

- Individual/Partner

TIMING

6 min

1. Follow the instructions in **starter_code_lesson_12 > this > js > main.js**

OBJECTS

OBJECTS

- OOP- Object oriented Programming
- Lets us write reusable code to keep track of data

Everything is an object!

BULLDOG AS AN OBJECT

Objects have traits that are common to versions of itself
These traits are called properties in javascript

Bulldog Properties

- Legs - 4
- Sound - “Bark”
- Food - “Dog Food"

DECLARING OBJECTS

```
function myObject(){

};
```

ASSIGNING PROPERTIES

```
function myObject(){
    this.property = value;
};
```

MAKING NEW OBJECTS

```
var newObject = new object();
```

GETTING OBJECT VALUES

```
var newObject = new object();

newObject.propertyName;
```

LAB

ACTIVITY — PANELS

KEY OBJECTIVE

- Apply programming skills to build a tab/panel widget

TIMING

12 min

1. Demo the panels site
2. Write pseudo code
3. Write JS

ACTIVITY — INTERACTIVE NAV

EXERCISE

KEY OBJECTIVE

- Apply programming skills to build an interactive nav

TYPE OF EXERCISE

- Individual/Partner

TIMING

8 min

1. Demo the interactive nav.
2. Write pseudo code

ACTIVITY — INTERACTIVE NAV

KEY OBJECTIVE

- Apply programming skills to build an interactive nav

TYPE OF EXERCISE

- Individual/Partner

TIMING

Until 9:15 1. Write JS to make interactive nav functional

HOMEWORK

HAVE CONTENT AND HTML SKELETON IN BOILERPLATE CODE READY THIS SUNDAY

FINISH INTERACTIVE NAV AND PANELS

EXIT TICKETS