LET'S GET EVERYTHING SET UP!

1. In Schoology, go to: **Courses(in the top menu) > FEWD CHI 1: Section 1**
2. Then go to the **Class Materials** folder — it's the pink one!
3. Navigate to the **Week 6 (It's the yellow folder) > Lesson 10 folder**
4. There you'll find all the materials for today's class
5. Download starter_code_lesson_10.zip
6. Move it from your Downloads folder to your Desktop
7. Double-click on starter_code_lesson_10.zip to unzip it
8. After you've unzipped, delete the original .zip to avoid confusion and make sure you don't unzip it again later!!!

GENERAL ASSEMBLY

ARRAYS

Sarah Holden

REVIEW

ADDING JQUERY TO YOUR WEBSITE — OPTION 1: JQUERY FILE

STEPS TO INCLUDE JQUERY:

1. Download the jQuery script 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>
  <!-- Content here -->
  <script src="js/jquery-1.11.2.min.js"></script>
  <script src="js/main.js"></script>
</body>
```

MAKE SURE THE DOCUMENT IS READY

- If you ever need to load jQuery in the head of your html file, we can use jQuery's ready method to detect when the DOM is ready to be manipulated.
- This waits for our HTML and CSS to load before executing

```
$(document).ready(function() {
  // your code here
});
```

=

Or use the shortcut:

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

***You **only** need to use jQuery's ready method when you're including your scripts in the head of your page.*

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
- We'll be using the shorthand in this class: $( )
- $(selector) is the same as jQuery(selector)

JQUERY OBJECTS — FINDING ELEMENTS: SOME EXAMPLES

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!

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:

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

JQUERY METHODS — GETTING/SETTING CONTENT

Get/change content of elements, attributes, text nodes

Some methods available to us:

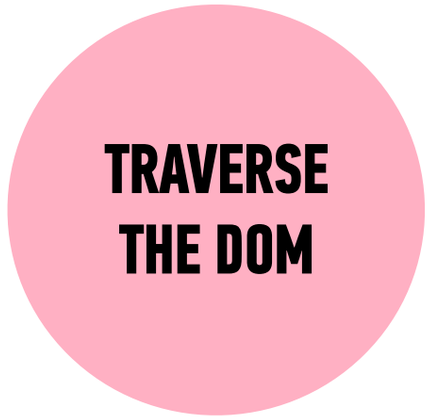
- .text()
- .html()
- .prepend()
- .append()
- .remove()
- .attr()
- .addClass()
- .removeClass()
- .css()

JQUERY METHODS — TRAVERSING THE DOM

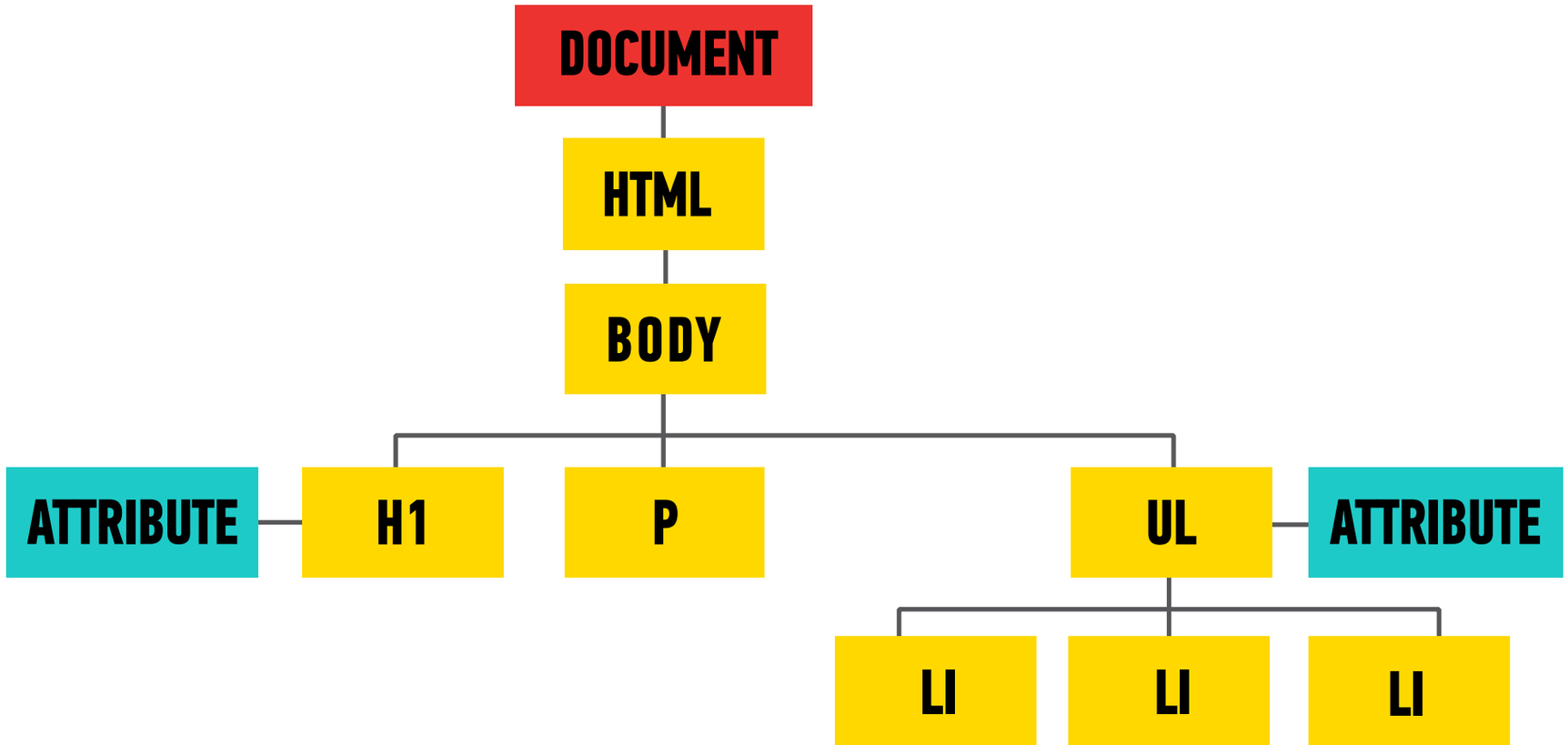
jQuery provides us with methods to find/select elements to work with & traverse the DOM

Some methods available to us:

- .find()
- .closest()
- .parent()
- .parents()
- .children()
- .siblings()
- .next()
- .nextAll()
- .prev
- .prevAll()



TRAVERSING THE DOM?



JQUERY METHODS — EFFECTS/ANIMATION

Add effects and animation to parts of the page

Some of the methods available to us:

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

JQUERY METHODS — EVENTS!

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

To add an event:

1. Use a selector to create a jQuery selection
2. Use .on() to indicate which event you want to respond to

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

Example:

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

CREATE EVENT LISTENERS

JQUERY METHODS — EVENTS!

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

- **UI:** focus, blur, change
- **Keyboard:** input, keydown, keyup
- **Mouse:** click, mouse, mousedown, mouseover
- **Form:** submit, select, change
- **Document:** ready, load
- **Browser:** resize, scroll

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

CREATE EVENT LISTENERS

JQUERY METHODS — THE EVENT OBJECT

- The event object has properties and methods that tell you more about the event that took place.
- We'll look at more later, but for now let's look at the preventDefault() method
- By using this method, the default action of the event will not be triggered.

Parameter name

```
$('li').on('eventGoesHere', function(e) {
  e.preventDefault();
});
```

Use that name in the function and use dot notation to access its properties and methods.

CREATE EVENT LISTENERS

HOMEWORK

HOMEWORK

Use e.preventDefault() to:

- Prevent a submit button from submitting a form
- Prevent a link from following the URL

USING JQUERY TO MANIPULATE THE DOM

1

Select an element/elements

2

Work with those elements

LEARNING OBJECTIVES

- Apply JS and jQuery knowledge to program a carousel.
- Define arrays
- Practice using indexes to access array elements

AGENDA

- Arrays
- Loops
- Lab

ARRAYS

STORING LISTS OF VALUES

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

DECLARING ARRAYS

There are 2 different ways to create an array:

ARRAY CONSTRUCTOR:

```
var myArr = new Array();
```

LITERAL NOTATION:

```
var myArr = [];
```

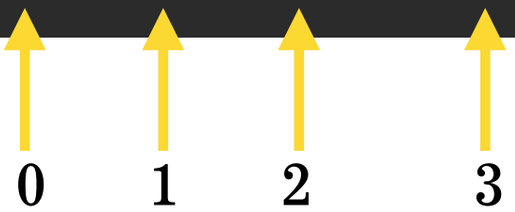
ARRAYS – INDEXING

- 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.

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



Accessing items in array:

`myArray[1]` => true

`myArray[2]` => 2

`myArray[0]` => 5

`myArray[3]` => 'Hello'

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 - INDEXING

- We can overwrite all the elements of an array simply by giving the array new values or by setting an array equal to a different array.

```
var fruits = ['Apples', 'Oranges', 'Pears', 'Bananas'];
var myArr = [1, 2, 3];
myArr = fruits;

console.log(myArr); // prints Apples, Oranges, Pears, Bananas
```

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'

CODE ALONG — ARRAYS

LOOPS

ITERATING OVER AN ARRAY

- The `forEach()` method loops through an array and applies the same function to every item in it.

```
var fruits = ['Banana', 'Apple', 'Pear'];
fruits.forEach(function(element, index){
    console.log(element, index);
});
```

***Element is the item from the array. Index is the item's position in the array. As always, code we want to execute goes between curly braces*

ARRAYS - MANY MORE METHODS

concat()	Joins two or more arrays, and returns a copy of the joined arrays
indexOf()	Search the array for an element and returns its position
join()	Joins all elements of an array into a string
lastIndexOf()	Search the array for an element, starting at the end, and returns its position
pop()	Removes the last element of an array, and returns that element
push()	Adds new elements to the end of an array, and returns the new length
reverse()	Reverses the order of the elements in an array

shift()	Removes the first element of an array, and returns that element
slice()	Selects a part of an array, and returns the new array
sort()	Sorts the elements of an array
splice()	Adds/Removes elements from an array
toString()	Converts an array to a string, and returns the result
unshift()	Adds new elements to the beginning of an array, and returns the new length
valueOf()	Returns the primitive value of an array

Find out more from W3Schools or MDN

CODE ALONG — LOOPS

LAB

ACTIVITY — IMAGE CAROUSEL

KEY OBJECTIVE

- Apply JS and jQuery knowledge to program a carousel.

TYPE OF EXERCISE

- Paired

TIMING

10 min	Write pseudo code

**Bonus*: *use the change event to give a ranking to the photos between 1 and 5. The user should be forwarded to the next image after voting.*

ACTIVITY — IMAGE CAROUSEL

KEY OBJECTIVE

- Apply JS and jQuery knowledge to program a carousel.

TYPE OF EXERCISE

- Paired

TIMING

Until 8:50	Implement with JS code

**Bonus*: *use the change event to give a ranking to the photos between 1 and 5. The user should be forwarded to the next image after voting.*

LEARNING OBJECTIVES

- Apply JS and jQuery knowledge to program a carousel.
- Define arrays
- Practice using indexes to access array elements

HOMEWORK

HOMEWORK

Starting this week, we'll be having everyone submit their homework on Schoology instead of via email!!

HOMEWORK

ASSIGNMENT:

Finish jQuery Carousel Lab - Due Saturday, February 21st at 11:30pm

REQUIRED READING:

From the textbook (JavaScript & jQuery by Jon Duckett):

- Chapter 10: Error Handling & Debugging

Otherwise:

- Read Javascript Debugging

OPTIONAL VIDEO:

Watch GA Front Row's Fundamentals of jQuery video

EXIT TICKETS