Review & jQuery

Learning Objectives

- **Review** JavaScript
- **Explain** what a library is
- **Explain** what jQuery is
- **Experiment** with downloading it
- **Recognize** jQuery's syntax
- **Practice** using it
- **Use** selectors and jQuery's function to effectively manipulate the DOM

Agenda

- Project Pitches
- Review
- jQuery
  - History
  - Downloading it
  - Using it

Project Pitches

Review

Data Types

```
"I am a string";

42;

true;

false;

undefined;

null;
```

Arithmetic Operators

```
+    -    PLUS

-    -    MINUS

*    -    MULTIPLY

/    -    DIVIDE

%    -    MODULO (REMAINDER)
```

Logical Operators

```
// &&     -     AND
// ||     -     OR
// !      -     NOT (Negation, opposite, or to boolean operator)

true && true;
4 < 5 && 5 >= 3;

true || false;
42 === 42 || 14 > 15;

!false;
!"Hello";
```

Comparison Operators

```
14 <= 15;

15 >= 11;

10 > 5;

20 < 42;

42 == "42"; // Type Coercion!

42 === 42;
```

If Conditionals

```
if ( CONDITION ) {   }

if ( CONDITION ) {

} else {

}

if ( CONDITION_ONE ) {

} else if ( CONDITION_TWO ) {

} else {

}
```

Functions

```
function sayHi () {
    console.log( "Say hi" );
}
sayHi();

function sayHello ( name ) {
    var msg = "Hello " + name;
    console.log( msg );
}
sayHello( "Elke" ); // "Hello Elke"
sayHello( "Zeppo" ); // "Hello Zeppo"
sayHello(); // "Hello undefined"

function add ( x, y ) {
    return x + y;
}
add( 4, 5 ); // => 9
```

Loops

```
// STARTING POINT, END CONDITION, STEP

for ( var i = 1; i <= 5; i += 1 ) {
    console.log( "I:", i );
}
```

Arrays

```
// Arrays
// Ordered, iterable, index list of any data
// Zero-based

var colors = [ "rebeccapurple", "lemonchiffon", "peachpuff" ];

colors[0]; // => "rebeccapurple"
colors[1]; // => "lemonchiffon"
colors[2]; // => "peachpuff"
colors[ colors.length - 1 ]; // => "peachpuff"

colors[0] = "chartreuse";
```

Iteration

```
var colors = [ "rebeccapurple", "lemonchiffon", "peachpuff" ];

for ( var i = 0; i < colors.length; i += 1 ) {
    var color = colors[ i ];
    console.log( color );
}
```

Objects

```
var bestMovie = {
    title: "Satantango",
    director: "Bela Tarr",
    duration: 432
};

// Accessing
bestMovie.duration; // => 432
bestMovie.title; // => "Satantango"

// Reassignment
bestMovie.duration = 534;
bestMovie.director = "Béla Tarr";

// Adding new properties
bestMovie.parts = 12;
bestMovie.language = "Hungarian";
```

Library

What is a library?

- A library is a collection of reusable functions for a particular purpose
- It's really just someone else's code that we use to do something. It could be:
  - A library of ways to easily access and manipulate the DOM
  - A library of really useless things, like this:
    - fartScroll.js
    - rekt.js
    - HTML9 Responsive BoilerStrap JS

How do we use them?

- We add a script that references their JS file above our own (to make sure that the code is available)
- How do we download them?
  - Local file
  - Content Delivery Network
- How do we find them?
  - cdnjs.com
  - JavaScripting.com
  - Google and browsing

jQuery

The Most Popular Library

- Released on August 22nd, 2005 by John Resig
- Current Statistics (as of 5-10-2017):
  - 18.6% of the entire internet uses it
  - 84.1% of the top million sites on the internet use it
  - 69,789,358 websites use it
- Found with Wappalyzer and BuiltWith

So, what does it do?

- Data manipulation
- DOM manipulation
- Events
- AJAX
- Effects and animation
- HTML templating
- Widgets / theming
- Graphics / charts
- App architecture
- Browser differences

jQuery

`Write less, do more`

How do we use it?

- jQuery defines a variable on our website called **$**
- We **select** a DOM Node with a **CSS-style selector** using that **$**
- We call a method on that node, or collection of nodes

What does it look like?

```
$( "img" ).fadeOut( 300 );
```

- $ refers to the function that jQuery defines for us
- "img" is our CSS Selector
- We call a method (jQuery provides the method) on the selected DOM nodes (all images)
- 300 is a parameter we provide to the fadeOut method (a time in milliseconds)

jQuery: In Depth

Selectors

```
$( "img" );

$( "p" );

$( ".bill" );

$( "#important" );

$( "div h3" );

$( "nav > a" );
```

Selectors

```
<p>Paragraph</p>

<div id="main"></div>

<p class="intro"></p>

<div class="container">
    <p></p>
</div>

<ul>
    <li></li>
    <li class="special">
        Special
    </li>
    <li></li>
    <li></li>
</ul>
```

```
$( "p" );

$( "#main" );

$( "p.intro" );


$( ".container p" );


$( "ul" );

$( "ul > li.special" );
```

Exercise!

Selectors

Reading Elements

```
<a href="http://jquery.com/"> jQuery </a>
```

```
$( "a" );

var text = $( "a" ).html();

var href = $( "a" ).attr( "href" );

var fontSize = $( "a" ).css( "font-size" );
```

Changing Elements

```
<a href="http://jquery.com/"> jQuery </a>
```

```
$( "a" );

$( "a" ).html( "MDN" );

$( "a" ).attr( "href", "https://developer.mozilla.org/en-US/" );

$( "a" ).css({
    fontSize: "50px"
}); // Notice the camelCasing!
```

Homework

- Try jQuery
- Code Academy: jQuery
- Learn jQuery
- jQuery Fundamentals

Next Lesson

- More jQuery

Q & A

Feedback Time

Lesson 12: **Review and jQuery**

https://ga.co/fewd32syd

Thanks!