

Introduction to jQuery

Our Goals

- Define what a library is in JavaScript
- A quick review
- Be able to talk about what jQuery actually is
- Get an introduction to how to use it

A quick review

```
var myString = "This is a string!";
var myNumber = 1.2;
var myBoolean = true;
var myArray = ["string", myNumber, {}, true];
var myObject = {
  name: "Pamela",
  adjective: "Cool",
  roles: ["hacker", "teacher", "coder"]
};
var myFunction = function (name) {
  console.log("Hello " + name);
}

myFunction("World!");
```

A quick review

```
document.getElementById( "id" );  
document.querySelector( "cssSelector" );  
  
document.getElementsByTagName( "tagName" );  
document.getElementsByClassName( "className" );  
document.querySelectorAll( "cssSelector" );  
  
var myButton = document.querySelector("button");  
myButton.addEventListener("click", function (e) {  
    console.log( e );  
});
```

What is a library?

- A library is a collection of reusable methods for a particular purpose
- It's really just someone else's code that we use to do something. It could be:
 - A library of maths-related things
 - 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 Bootstrap JS](#)

A math library...

- Might include things like:
 - ***math.sum(2, 5);***
 - ***math.pow(4, 6);***
 - ***math.factorial(9);***
- To work with a javascript library:
 - We include the library before our own as a script tag
 - Then we use it in our code
- [Like this!](#)

Have a crack at **these**
exercises

jQuery: The most popular library

- Built in August 22nd, 2005 by John Resig
 - [Twitter](#)
 - [Github](#)
- 11.5% of the entire internet use it
- 78.1% of the top million sites use it
- 41, 691, 337 sites use it (the last I checked)
 - [Wappalyzer](#)
 - [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

So, why jQuery?

```
// Pure JavaScript version
```

```
var images = document.getElementsByClassName( "img" );
```

```
for (var i = 0; i < images.length; i++) {  
    var currentImage = images[i];  
    currentImage.style.display = "none";  
}
```

```
// jQuery version
```

```
$("img").hide(); // Or...
```

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

So, why jQuery?

```
var p = document.createElement('p');
var textNode = document.createTextNode('Welcome!');
p.appendChild(textNode);
p.style.cssFloat = 'left';
p.style.backgroundColor = 'red';
p.className = 'special';
var header = document.getElementsByClassName('header')[0];
header.appendChild( p );
```

// jQuery version

```
var $p = $("<p>Welcome!</p>");
$p.css({ float: "left", backgroundColor: "red" });
$p.addClass( "special" );
$("header").append( $p );
```

Select, manipulate, admire!

```
// Select all paragraphs on the page
var $allParagraphs = $( "p" );

// Use methods on them all
$allParagraphs.addClass( "special" );
$allParagraphs.css( "font-size", "24px" );
$allParagraphs.fadeOut();

// Admiring goes here...
```

jQuery: The Basics

```
$( "p" ).addClass( "special" );
```

- **\$** - The global jQuery function. Can also be "jQuery"
- **("p")** - Finds DOM element(s) according to what's in the quotes. Returns a "jQuery collection."
- **.addClass("special");** - Built-in jQuery method that adds the specified class to the collection. [Read the docs here.](#)

jQuery: The Basics

All CSS selectors are valid, plus [heaps more!](#)

```
<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 li.special" );
```

Have a crack at **these**
exercises

Reading elements

```
<a href="http://ga.co" style="font-size:20px;">GA</a>
```

```
// We can store it
var $aTag = $( "a" );

var aTagHTML = $aTag.html();
var aHref = $aTag.attr("href");
var aFontSize = $aTag.css("font-size");
```


Changing elements

```
<a href=" "></a>
```

```
var $aTag = $( 'a' );  
  
$aTag.html( 'Yahoo!' );  
$aTag.attr( 'href', 'http://www.yahoo.com' );  
$aTag.css({  
    'color': 'hotpink'  
});
```

```
<a href="http://yahoo.com" style="color:hotpink;">Yahoo!</a>
```

Create, manipulate, inject!

- **Step 1:** Create element and store a reference
 - `var $p = $("<p></p>");`
- **Step 2:** Use a method to manipulate (optional)
 - `$p.addClass("special");`
- **Step 3:** Inject into your HTML
 - `$("body").append($p);`

Create, manipulate, inject!

- **Step 1:** Create element and store a reference
 - `var $h1 = $("<h1>Hello World</h1>");`
- **Step 2:** Manipulate the CSS
 - `$h1.css("font-size", "24px");`
- **Step 3:** Make sure you put it on the page!
 - `$("body").prepend($h1);`

Regular Nodes to jQuery

```
var $allParagraphs = $("p"); // jQuery collection
var firstParagraph = $allParagraphs[0]; // Pure DOM
var $firstParagraph = $allParagraphs.eq(0); // jQuery
for (var i = 0; i < $allParagraphs.length; i++) {
    var $paragraph = $allParagraphs.eq( i );
    $paragraph.html( $paragraph.html() += " wooooo!" );
}
```

Iterating through (proper)

```
var $allParagraphs = $( "<p>" );  
  
$allParagraphs.each(function () {  
    $(this).html( "Wow!" );  
});
```

jQuery: Including

- Download it from [here](#)
- Or link to the official CDN

```
<script src="https://code.jquery.com/jquery-1.12.3.min.js"></script>

<!-- OR... -->

<script src="jquery.js"></script>

<!-- Then your code! -->

<script src="main.js"></script>
```

Have a crack at **these**
exercises

Our Goals

- Define what a library is in JavaScript
- A quick review
- Be able to talk about what jQuery actually is
- Get an introduction to how to use it

Homework

- [Dancing Cats](#)
- [Try jQuery](#)
- [Code Academy: jQuery](#)
- [Learn jQuery](#)
- [jQuery Fundamentals](#)