Intro to jQuery

`http://jquery.com`

jQuery?

Pros

- Makes code simpler
- Easier to read means cheaper and more reliable
- Evens out differences between browsers

Cons

- Masks what's "really" going on
- Can lead to *cargo cult coding* (putting code you don't understand in your app)
- *May* become irrelevant as all browsers support the same JavaScript API and Document & Browser Object Models

Add jQuery library to your site

- Get it minified from a Content Delivery Service (CDN). Faster for your customers.

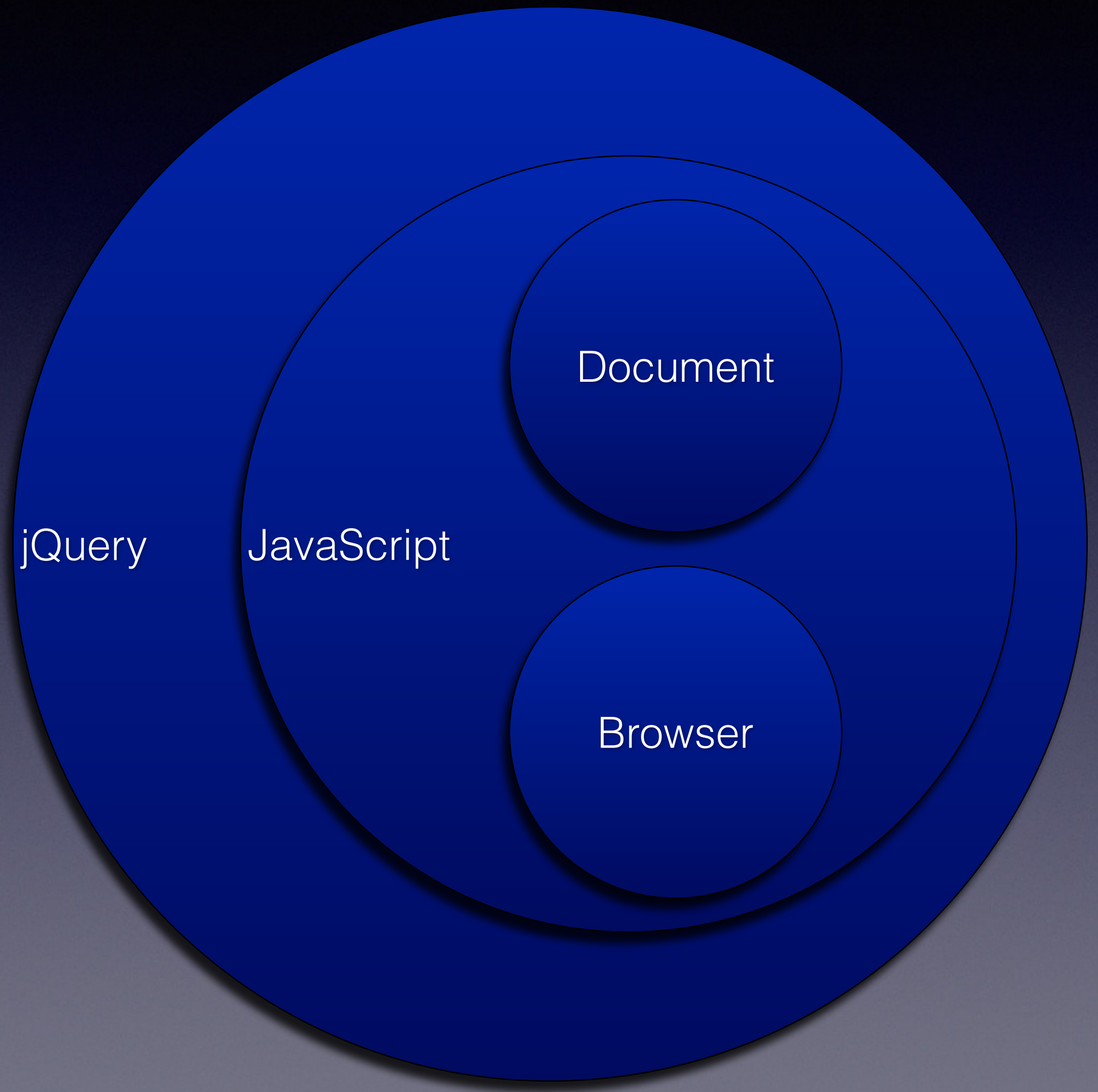
```
<script src=“http://ajax.googleapis.com/ajax/libs/jquery/2.0.3/jquery.min.js”>
</script>
```

- Download a local copy & serve it yourself. Allows reading & debugging the code.

```
<script src=“scripts/jquery-2.0.3.js"></script>
```

Basic Concepts

- jQuery is an API to the JavaScript API to the Document & Browser
- While you can do everything you need in JavaScript,
- jQuery syntax makes it easier to code & read
- jQuery leverages community contributions via “plugins”



Simpler Syntax

`document.getElementByClassName('container')`	`$('.container')`
`document.getElementById('theTable')`	`$('#theTable')`
`document.getElementByTagName('p')`	`$('p')`

Basic Operations

- **Selection** - Pick something based on a CSS selector
- **Traversal** - Pick some thing relative to some other thing
- **Manipulation** - Change the thing you've selected
- **Events** - Respond to events to select and manipulate stuff
- **Method chaining** - calling functions one after another

Selection

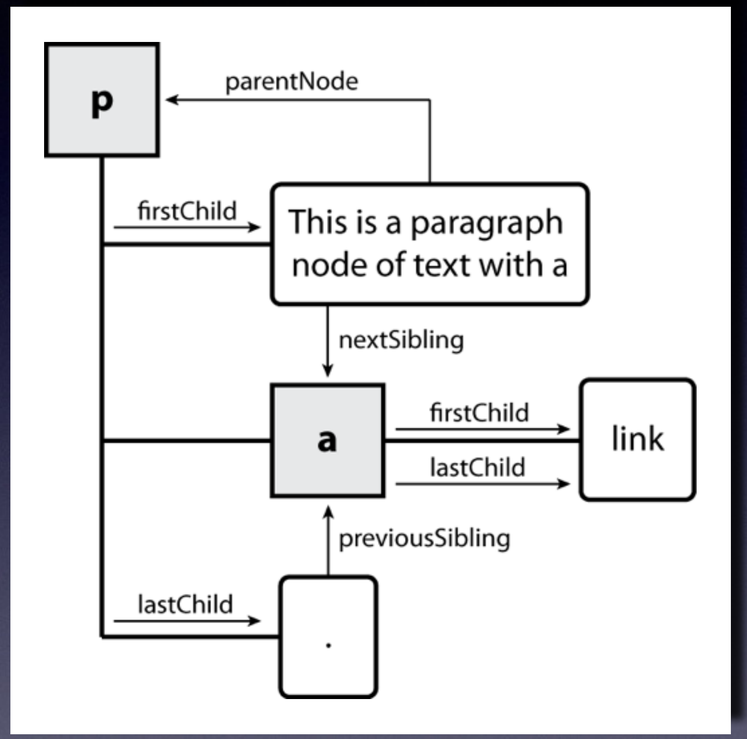
- Element type
- Element attribute
- Element class
- Element ID
- Pseudo classes

The selection criteria used by jQuery is essentially the same as used by CSS.

You only need to learn the selection language once to use both CSS and jQuery.

Traversal

- Starts with the results of a selection
- Moves through the DOM
- Returns a new set of elements



Example

```
<p>This is a paragraph node with a <a>link</a>.
```

Manipulation

- Add to or change content of the selected elements
- Insert new elements before or after selected elements
- Animate selected elements (move, reveal, highlight, etc.)

Example: Change element style by changing element class

```
$('[required =“required”]').addClass('required'); // in JS file

.required {color: red}                              // in CSS file
```

Method Chaining

Automatically feeds the return value of one method (usually an array of elements) to the next method in the chain

```
$('[required ="required"]').prev('label').append('<span
>*</span>').children('span').addClass('required');
```

```
$('[required =“required”]')// find all elements with a “required” attribute
.prev('label')              // traverse backwards to previous <label> elements
.append('<span >*</span>') // append a <span> element
.children('span')           // find all children of the <span> elements
.addClass('required');      // add a class so we can style them differently
```

Events

- **document loaded** - DOM is built and can be manipulated
- **window changes** - user is looking elsewhere
- **error occurs** - something went wrong
- **click** - mouse button pressed & released
- **doubleclick** - element is selected
- **hover**: mouse hovers over the element.
- **mousedown**: mouse button is pressed, and before the button is released.
- **mouseup**: mouse button is released.
- **keypress**: each time the user types a key into an element, such as a text field.
- **keydown**: user presses a key, but before the output is reflected on screen. This allows you to veto the action
- **focus**: the form field receives focus.
- **blur**: focus leaves a form field. Used to validate content after the user has finished typing.
- **change**: the value of a form field changes.

- Cameron, Dane (2013-10-30). A Software Engineer Learns HTML5, JavaScript and jQuery: A guide to standards-based web applications (p. 111). Cisdal Publishing. Kindle Edition.