

jQuery®

Notes for Professionals

Chapter 2: Selectors

A jQuery selector selects or finds a DOM (document object model) element in an HTML document. It selects HTML elements based on ID, name, types, attributes, class and etc. It is based on existing elements.

Section 2.1: Overview

Elements can be selected by jQuery using [jQuery Selectors](#). The function returns either an array of elements.

```
Basic selectors
$( "*" ) // All elements
$( "div" ) // All div elements
$( ".div" ) // All elements with class=div AND class=red
$( "#div" ) // All elements with class=div OR class=red
$( "div, red" ) // All elements with class=div OR class=red
$( "div, #red" ) // The first element with id=red
$( "#div" ) // All elements with id=div
$( "a[href]" ) // All elements with href=example.com
$( "a[href=example.com]" ) // All elements with href=example.com
```

```
Relational operators
$( "div > div" ) // All divs that are descendants of a div
$( "div ~ div" ) // All divs that are siblings following an div
$( "div + div" ) // All divs that are immediately after an div
$( "#a ~ #b" ) // All divs that are immediately after an div
```

Section 2.2: Types of Selectors

In jQuery you can select elements in a page using many various properties of the

- Type
- Class
- ID
- Possession of Attribute
- Attribute Value
- Indexed Selector
- pseudo-selector

If you know CSS selectors you will notice selectors in jQuery are the same. Take the following HTML, for example:

```
<a href="#link" id="1" >1</a>
<a id="second-link" >2</a>
<a class="example" >3</a>
<a class="example" href="#out.html" >4</a>
<a class="example" >5</a>
```

Selecting by Type: The following jQuery selector will select all <a> elements, including 1, 2, 3 and 4.

```
$( "a" )
```

Chapter 7: DOM Manipulation

Section 7.1: Creating DOM elements

The `jQuery` function (usually aliased as `$`) can be used both to select elements and to create new elements.

```
var myLink = $( "<a href='http://stackoverflow.com'/>JS" );
```

You can optionally pass a second argument with element attributes:

```
var myLink = $( "<a href='http://stackoverflow.com'/>" );
```

`<a>` → The first argument specifies the type of DOM element you want to create. In this example it's an `anchor`, but could be anything [on this list](#). See the [specification](#) for a reference of the `<a>` element.

`{ href: "http://stackoverflow.com" }` → The second argument is a [JavaScript Object](#) containing attribute name/value pairs.

The 'name' value pairs will appear between the `<` of the first argument, for example `` which for our example would be ``.

Section 7.2: Manipulating element classes

Assuming the page includes an HTML element like:

```
<p class="small-paragraph">
  This is a small <a href="https://en.wikipedia.org/wiki/Paragraph">paragraph</a>
  with a <a class="trusted" href="http://stackoverflow.com">link</a> inside.
</p>
```

jQuery provides useful functions to manipulate DOM classes, most notably `hasClass()`, `addClass()`, `removeClass()` and `toggleClass()`. These functions directly modify the class attribute of the matched elements.

```
$( "#p" ).hasClass( "small-paragraph" ); // true
$( "#p" ).hasClass( "large-paragraph" ); // false
// Add a class to all links within paragraphs
$( "#p a" ).addClass( "trusted-link-in-paragraph" );
```

```
// Remove the class from a trusted
$( "#trusted-untreated-link-in-paragraph" )
  .removeClass( "trusted-link-in-paragraph" )
  .addClass( "trusted-link-in-paragraph" );
```

Toggle a class

Given the example markup, we can add a class with our first `toggleClass()`:

```
$( "#small-paragraph" ).toggleClass( "pretty" );
```

Now this would return true: `$("#small-paragraph").hasClass("pretty")`

`toggleClass` provides the same effect with less code as:

```
if( $( "#small-paragraph" ).hasClass( "pretty" ) )
```

jQuery® Notes for Professionals

Chapter 10: Element Visibility

Parameter When passed, the effects of `.hide()`, `.show()` and `.toggle()` are animated; the elements will gradually fade in or out. Details

Section 10.1: Overview

```
$(element).hide() // sets visibility: none
$(element).show() // sets visibility to original value
$(element).toggle() // toggles between the two
$(element).is(':visible') // returns true or false
$(element).is(':hidden') // returns all elements that are visible
$(element).fadeOut() // displays the element
$(element).fadeIn() // hides the element
$(element).fadeIn(1000) // display the element using timer
$(element).fadeOut(1000) // hide the element using timer
// code to execute
// hide the element using timer and a callback function
$(element).fadeOut(1000, function() {
  // code to execute
});
```

Section 10.2: Toggle possibilities

Simple toggle() case

```
function toggleBasic() {
  $( "#target" ).toggle();
}
```

With specific duration

```
function toggleDuration() {
  $( "#target" ).toggle( "slow" ); // A 400ms duration value is also acceptable
```

...and callback

```
function toggleCallback() {
  $( "#target" ).toggle( "slow", function() {alert( "now do something" );});
```

...or with easing and callback.

```
function toggleEasingCallback() {
  // The new jQuery 1.9+ only supports linear and swing easings
  $( "#target" ).toggle( "slow", "linear", function() {alert( "now do something" );});
}
```

50+ pages
of professional hints and tricks

Contents

About	1
Chapter 1: Getting started with jQuery	2
Section 1.1: Getting Started	2
Section 1.2: Avoiding namespace collisions	3
Section 1.3: jQuery Namespace ("jQuery" and "\$")	4
Section 1.4: Loading jQuery via console on a page that does not have it	5
Section 1.5: Include script tag in head of HTML page	5
Section 1.6: The jQuery Object	7
Chapter 2: Selectors	8
Section 2.1: Overview	8
Section 2.2: Types of Selectors	8
Section 2.3: Caching Selectors	10
Section 2.4: Combining selectors	11
Section 2.5: DOM Elements as selectors	13
Section 2.6: HTML strings as selectors	13
Chapter 3: Each function	15
Section 3.1: jQuery each function	15
Chapter 4: Attributes	16
Section 4.1: Difference between attr() and prop()	16
Section 4.2: Get the attribute value of a HTML element	16
Section 4.3: Setting value of HTML attribute	17
Section 4.4: Removing attribute	17
Chapter 5: document-ready event	18
Section 5.1: What is document-ready and how should I use it?	18
Section 5.2: jQuery 2.2.3 and earlier	18
Section 5.3: jQuery 3.0	19
Section 5.4: Attaching events and manipulating the DOM inside ready()	19
Section 5.5: Difference between \$(document).ready() and \$(window).load()	20
Section 5.6: Difference between jQuery(fn) and executing your code before </body>	21
Chapter 6: Events	22
Section 6.1: Delegated Events	22
Section 6.2: Attach and Detach Event Handlers	23
Section 6.3: Switching specific events on and off via jQuery. (Named Listeners)	24
Section 6.4: originalEvent	25
Section 6.5: Events for repeating elements without using ID's	25
Section 6.6: Document Loading Event .load()	26
Chapter 7: DOM Manipulation	27
Section 7.1: Creating DOM elements	27
Section 7.2: Manipulating element classes	27
Section 7.3: Other API Methods	29
Chapter 8: DOM Traversing	31
Section 8.1: Select children of element	31
Section 8.2: Get next element	31
Section 8.3: Get previous element	31
Section 8.4: Filter a selection	32
Section 8.5: find() method	33

Section 8.6: Iterating over list of jQuery elements	34
Section 8.7: Selecting siblings	34
Section 8.8: closest() method	34
Chapter 9: CSS Manipulation	36
Section 9.1: CSS – Getters and Setters	36
Section 9.2: Increment/Decrement Numeric Properties	36
Section 9.3: Set CSS property	37
Section 9.4: Get CSS property	37
Chapter 10: Element Visibility	38
Section 10.1: Overview	38
Section 10.2: Toggle possibilities	38
Chapter 11: Append	40
Section 11.1: Efficient consecutive .append() usage	40
Section 11.2: jQuery append	43
Section 11.3: Appending an element to a container	43
Chapter 12: Prepend	45
Section 12.1: Prepending an element to a container	45
Section 12.2: Prepend method	45
Chapter 13: Getting and setting width and height of an element	47
Section 13.1: Getting and setting width and height (ignoring border)	47
Section 13.2: Getting and setting innerWidth and innerHeight (ignoring padding and border)	47
Section 13.3: Getting and setting outerWidth and outerHeight (including padding and border)	47
Chapter 14: jQuery .animate() Method	48
Section 14.1: Animation with callback	48
Chapter 15: jQuery Deferred objects and Promises	50
Section 15.1: jQuery ajax() success, error VS .done(), .fail()	50
Section 15.2: Basic promise creation	50
Chapter 16: Ajax	52
Section 16.1: Handling HTTP Response Codes with \$.ajax()	52
Section 16.2: Using Ajax to Submit a Form	53
Section 16.3: All in one examples	53
Section 16.4: Ajax File Uploads	55
Chapter 17: Checkbox Select all with automatic check/uncheck on other checkbox change	58
Section 17.1: 2 select all checkboxes with corresponding group checkboxes	58
Chapter 18: Plugins	59
Section 18.1: Plugins - Getting Started	59
Credits	61
You may also like	64

About

Please feel free to share this PDF with anyone for free,
latest version of this book can be downloaded from:
<https://goalkicker.com/jqueryBook>

This *jQuery® Notes for Professionals* book is compiled from [Stack Overflow Documentation](#), the content is written by the beautiful people at Stack Overflow. Text content is released under Creative Commons BY-SA, see credits at the end of this book whom contributed to the various chapters. Images may be copyright of their respective owners unless otherwise specified

This is an unofficial free book created for educational purposes and is not affiliated with official jQuery® group(s) or company(s) nor Stack Overflow. All trademarks and registered trademarks are the property of their respective company owners

The information presented in this book is not guaranteed to be correct nor accurate, use at your own risk

Please send feedback and corrections to web@petercv.com

Chapter 1: Getting started with jQuery

Version	Notes	Release Date
1.0	First stable release	2006-08-26
1.1		2007-01-14
1.2		2007-09-10
1.3	Sizzle introduced into core	2009-01-14
1.4		2010-01-14
1.5	Deferred callback management, ajax module rewrite	2011-01-31
1.6	Significant performance gains in the <code>attr()</code> and <code>val()</code> methods	2011-05-03
1.7	New Event APIs: <code>on()</code> and <code>off()</code> .	2011-11-03
1.8	Sizzle rewritten, improved animations and <code>\$(html, props)</code> flexibility.	2012-08-09
1.9	Removal of deprecated interfaces and code cleanup	2013-01-15
1.10	Incorporated bug fixes and differences reported from both the 1.9 and 2.0 beta cycles	2013-05-24
1.11		2014-01-24
1.12		2016-01-08
2.0	Dropped IE 6–8 support for performance improvements and reduction in size	2013-04-18
2.1		2014-01-24
2.2		2016-01-08
3.0	Massive speedups for some jQuery custom selectors	2016-06-09
3.1	No More Silent Errors	2016-07-07
3.2	No More Silent Errors	2017-03-16
3.3	No More Silent Errors	2018-01-19

Section 1.1: Getting Started

Create a file `hello.html` with the following content:

```
<!DOCTYPE html>
<html>
<head>
  <title>Hello, World!</title>
</head>
<body>
  <div>
    <p id="hello">Some random text</p>
  </div>
  <script src="https://code.jquery.com/jquery-2.2.4.min.js"></script>
  <script>
    $(document).ready(function() {
      $('#hello').text('Hello, World!');
    });
  </script>
</body>
</html>
```

[Live Demo on JSBin](#)

Open this file in a web browser. As a result you will see a page with the text: `Hello, World!`

Explanation of code

[Click here to download full PDF material](#)