

FINDING ELEMENTS

Using jQuery, you usually select elements using CSS-style selectors. It also offers some extra selectors, noted below with a 'jQ'.

Examples of using these selectors are demonstrated throughout the chapter. The syntax will be familiar to those who have used selectors in CSS.

BASIC SELECTORS

*	All elements
<i>element</i>	All elements with that element name
<i>#id</i>	Elements whose id attribute has the value specified
<i>.class</i>	Elements whose class attribute has the value specified
<i>selector1, selector2</i>	Elements that match more than one selector (see also the .add() method, which is more efficient when combining selections)

HIERARCHY

<i>ancestor descendant</i>	An element that is a descendant of another element (e.g., li a)
<i>parent > child</i>	An element that is a direct child of another element (you can use * in the place of the child to select all child elements of the specified parent)
<i>previous + next</i>	Adjacent sibling selector only selects elements that are immediately followed by the previous element
<i>previous ~ siblings</i>	Sibling selector will select any elements that are a sibling of the previous element

BASIC FILTERS

<i>:not(selector)</i>	All elements except the one in the selector (e.g., div:not('#summary'))
<i>:first</i>	jQ The first element from the selection
<i>:last</i>	jQ The last element from the selection
<i>:even</i>	jQ Elements with an even index number in the selection
<i>:odd</i>	jQ Elements with an odd index number in the selection
<i>:eq(index)</i>	jQ Elements with an index number equal to the one in the parameter
<i>:gt(index)</i>	jQ Elements with an index number greater than the parameter
<i>:lt(index)</i>	jQ Elements with an index number less than the parameter
<i>:header</i>	jQ All <h1> - <h6> elements
<i>:animated</i>	jQ Elements that are currently being animated
<i>:focus</i>	The element that currently has focus

CONTENT FILTERS

:contains('text')		Elements that contain the specified text as a parameter
:empty		All elements that have no children
:parent	jQ	All elements that have a child node (can be text or element)
:has(selector)	jQ	Elements that contain at least one element that matches the selector (e.g., div:has(p) matches all div elements that contain a <p> element)

VISIBILITY FILTERS

:hidden	jQ	All elements that are hidden
:visible	jQ	All elements that consume space in the layout of the page Not selected if: display: none; height / width: 0; ancestor is hidden Selected if: visibility: hidden; opacity: 0 because they would take up space in layout

CHILD FILTERS

:nth-child(expr)		The value here is not zero-based e.g. ul li:nth-child(2)
:first-child		First child from the current selection
:last-child		Last child from the current selection
:only-child		When there is only one child of the element (div p:only-child)

ATTRIBUTE FILTERS

[attribute]		Elements that carry the specified attribute (with any value)
[attribute='value']		Elements that carry the specified attribute with the specified value
[attribute!='value']	jQ	Elements that carry the specified attribute but not the specified value
[attribute^='value']		The value of the attribute begins with this value
[attribute\$='value']		The value of the attribute ends with this value
[attribute*='value']		The value should appear somewhere in the attribute value
[attribute ='value']		Equal to given string, or starting with string and followed by a hyphen
[attribute~='value']		The value should be one of the values in a space separated list
[attribute][attribute2]		Elements that match all of the selectors

FORM

:input	jQ	All input elements
:text	jQ	All text inputs
:password	jQ	All password inputs
:radio	jQ	All radio buttons
:checkbox	jQ	All checkboxes
:submit	jQ	All submit buttons
:image	jQ	All elements
:reset	jQ	All reset buttons
:button	jQ	All <button> elements
:file	jQ	All file inputs
:selected	jQ	All selected items from drop-down lists
:enabled		All enabled form elements (the default for all form elements)
:disabled		All disabled form elements (using the CSS disabled property)
:checked		All checked radio buttons or checkboxes

DOING THINGS WITH YOUR SELECTION

Once you have seen the basics of how jQuery works, most of this chapter is dedicated to demonstrating these methods.

These two pages both offer an overview to the jQuery methods and will also help you find the methods you are looking for once you have read the chapter.

You often see jQuery method names written starting with a period (.) before the name. This convention is used in this book to help you easily identify those methods as being jQuery methods rather than built-in JavaScript methods, or methods of custom objects.

When you make a selection, the **jQuery** object that is created has a property called **length**, which will return the number of elements in the object.

If the jQuery selection did not find any matching elements, you will not get an error by calling any of these methods – they just won't do or return anything.

There are also methods that are specifically designed to work with Ajax (which lets you refresh part of the page rather than an entire page) shown in Chapter 8.

CONTENT FILTERS

Get or change content of elements, attributes, text nodes

GET/CHANGE CONTENT

<code>.html()</code>	p316
<code>.text()</code>	p316
<code>.replaceWith()</code>	p316
<code>.remove()</code>	p316

ELEMENTS

<code>.before()</code>	p318
<code>.after()</code>	p318
<code>.prepend()</code>	p318
<code>.append()</code>	p318
<code>.remove()</code>	p346
<code>.clone()</code>	p346
<code>.unwrap()</code>	p346
<code>.detach()</code>	p346
<code>.empty()</code>	p346
<code>.add()</code>	p338

ATTRIBUTES

<code>.attr()</code>	p320
<code>.removeAttr()</code>	p320
<code>.addClass()</code>	p320
<code>.removeClass()</code>	p320
<code>.css()</code>	p322

FORM VALUES

<code>.val()</code>	p343
<code>.isNumeric()</code>	p343

FINDING ELEMENTS

Find and select elements to work with & traverse the DOM

GENERAL

<code>.find()</code>	p336
<code>.closest()</code>	p336
<code>.parent()</code>	p336
<code>.parents()</code>	p336
<code>.children()</code>	p336
<code>.siblings()</code>	p336
<code>.next()</code>	p336
<code>.nextAll()</code>	p336
<code>.prev()</code>	p336
<code>.prevAll()</code>	p336

FILTER/TEST

<code>.filter()</code>	p338
<code>.not()</code>	p338
<code>.has()</code>	p338
<code>.is()</code>	p338
<code>:contains()</code>	p338

ORDER IN SELECTION

<code>.eq()</code>	p340
<code>.lt()</code>	p340
<code>.gt()</code>	p340

Once you have selected the elements you want to work with (and they are in a jQuery object), the jQuery methods listed on these two pages perform tasks on those elements.

DIMENSION/POSITION

Get or update the dimensions or position of a box

DIMENSION

<code>.height()</code>	p348
<code>.width()</code>	p348
<code>.innerHeight()</code>	p348
<code>.innerWidth()</code>	p348
<code>.outerHeight()</code>	p348
<code>.outerWidth()</code>	p348
<code>\$(document).height()</code>	p350
<code>\$(document).width()</code>	p350
<code>\$(window).height()</code>	p350
<code>\$(window).width()</code>	p350

POSITION

<code>.offset()</code>	p351
<code>.position()</code>	p351
<code>.scrollLeft()</code>	p350
<code>.scrollTop()</code>	p350

EFFECTS & ANIMATION

Add effects and animation to parts of the page

BASIC

<code>.show()</code>	p332
<code>.hide()</code>	p332
<code>.toggle()</code>	p332

FADING

<code>.fadeIn()</code>	p332
<code>.fadeOut()</code>	p332
<code>.fadeTo()</code>	p332
<code>.fadeToggle()</code>	p332

SLIDING

<code>.slideDown()</code>	p332
<code>.slideUp()</code>	p332
<code>.slideToggle()</code>	p332

CUSTOM

<code>.delay()</code>	p332
<code>.stop()</code>	p332
<code>.animate()</code>	p332

EVENTS

Create event listeners for each element in the selection

DOCUMENT/FILE

<code>.ready()</code>	p312
<code>.load()</code>	p313

USER INTERACTION

<code>.on()</code>	p326
There used to be methods for individual types of event, so you may see methods such as <code>.click()</code> , <code>.hover()</code> , <code>.submit()</code> . However, these have been dropped in favour of the <code>.on()</code> method to handle events.	