

## LEGEND

<i>El</i>	Element	<b>a</b> <>	Array
<i>F<sub>x</sub></i>	Function	<b>a</b> < <i>i</i> >	Array of <i>i</i>
<i>Num</i>	Number	<b>a</b> < <i>i</i> ( <b>s</b> )>	Array of <i>i</i> ( <b>s</b> )
<i>Int</i>	Integer	<b>T</b>	Notes
0-1	Boolean	*	Deprecated in jQuery 1.3

Designed by Antonio Lupetti  
 • <http://woork.blogspot.com> • <http://www.twitter.com/woork>

## ★ SELECTORS / 1. BASIC

<b>#id</b>	Matches a single element with the given id attribute.	<b>a</b> < <i>El</i> >
<b>element</b>	Matches all elements with the given name.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>.class</b>	Matches all elements with the given class.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>.class.class</b>	Matches all elements with the given classes.	<b>a</b> < <i>El</i> ( <b>s</b> )>
*	Matches all elements.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>selector1, selector2, selectorN</b>	Matches the combined results of all the specified selectors.	<b>a</b> < <i>El</i> ( <b>s</b> )>

## ★ SELECTORS / 2. HIERARCHY

<b>ancestor descendant</b>	Matches all descendant elements specified by "descendant" of elements specified by "ancestor".	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>parent &gt; child</b>	Matches all child elements specified by "child" of elements specified by "parent".	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>prev + next</b>	Matches all next elements specified by "next" that are next to elements specified by "prev".	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>prev ~ siblings</b>	Matches all sibling elements after the "prev" element that match the filtering "siblings" selector.	<b>a</b> < <i>El</i> ( <b>s</b> )>

## ★ SELECTORS / 3. BASIC FILTERS

<b>:first</b>	Matches the first selected element.	<b>a</b> < <i>El</i> >
<b>:last</b>	Matches the last selected element.	<b>a</b> < <i>El</i> >
<b>:not(selector)</b>	Filters out all elements matching the given selector.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:even</b>	Matches even elements, zero-indexed.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:odd</b>	Matches odd elements, zero-indexed.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:eq(index)</b>	Matches a single element by its index.	<b>a</b> < <i>El</i> >
<b>:gt(index)</b>	Matches all elements with an index above the given one.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:lt(index)</b>	Matches all elements with an index below the given one.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:header</b>	Matches all elements that are headers, like h1, h2, h3 and so on.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:animated</b>	Matches all elements that are currently being animated.	<b>a</b> < <i>El</i> ( <b>s</b> )>

## ★ SELECTORS / 4. CONTENT FILTERS

<b>:contains(text)</b>	Matches elements which contain the given text.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:empty</b>	Matches all elements that have no children (including text nodes).	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:has(selector)</b>	Matches elements which contain at least one element that matches the specified selector.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:parent</b>	Matches all elements that are parents - they have child elements, including text.	<b>a</b> < <i>El</i> ( <b>s</b> )>

## ★ SELECTORS / 5. ATTRIBUTE FILTERS

<b>[attribute]</b>	Matches elements that have the specified attribute.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>[attribute=value]</b>	Matches elements that have the specified attribute with a certain value.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>[attribute!=value]</b>	Matches elements that either don't have the specified attribute or do have the specified attribute but not with a certain value.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>[attribute^=value]</b>	Matches elements that have the specified attribute and it starts with a certain value.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>[attribute\$=value]</b>	Matches elements that have the specified attribute and it ends with a certain value.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>[attribute*=value]</b>	Matches elements that have the specified attribute and it contains a certain value.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>[attributeFilter1][attributeFilter2][attributeFilterN]</b>	Matches elements that match all of the specified attribute filters.	<b>a</b> < <i>El</i> ( <b>s</b> )>

## ★ SELECTORS / 6. CHILD FILTERS

<b>:nth-child(index/even/odd/equation)</b>	Matches all elements that are the nth-child of their parent or that are the parent's even or odd children.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:first-child</b>	Matches all elements that are the first child of their parent.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:last-child</b>	Matches all elements that are the last child of their parent.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:only-child</b>	Matches all elements that are the only child of their parent.	<b>a</b> < <i>El</i> ( <b>s</b> )>

## ★ SELECTORS / 7. VISIBILITY FILTERS

<b>:hidden</b>	Matches all elements that are hidden.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:visible</b>	Matches all elements that are visible.	<b>a</b> < <i>El</i> ( <b>s</b> )>

## ★ SELECTORS / 8. FORMS

<b>:input</b>	Matches all input, textarea, select and button elements.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:text</b>	Matches all input elements of type text.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:password</b>	Matches all input elements of type password.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:radio</b>	Matches all input elements of type radio.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:checkbox</b>	Matches all input elements of type checkbox.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:submit</b>	Matches all input elements of type submit.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:image</b>	Matches all input elements of type image.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:reset</b>	Matches all input elements of type reset.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:button</b>	Matches all button elements and input elements of type button.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:file</b>	Matches all input elements of type file.	<b>a</b> < <i>El</i> ( <b>s</b> )>

## ★ SELECTORS / 9. FORM FILTERS

<b>:enabled</b>	Matches all elements that are enabled.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:disabled</b>	Matches all elements that are disabled.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:checked</b>	Matches all elements that are checked.	<b>a</b> < <i>El</i> ( <b>s</b> )>
<b>:selected</b>	Matches all elements that are selected.	<b>a</b> < <i>El</i> ( <b>s</b> )>

★ CORE / 1. THE JQUERY FUNCTION

**jQuery( expression, context )**

This function accepts a string containing a CSS selector which is then used to match a set of elements.

jQuery

**jQuery( html, ownerDocument )**

Create DOM elements on-the-fly from the provided String of raw HTML.

jQuery

**jQuery( elements )**

Wrap jQuery functionality around a single or an array of DOM Element(s).

jQuery

**jQuery( callback )**

A shorthand for \$(document).ready().

jQuery

★ CORE / 2. OBJECT ACCESSORS

**each( callback )**

Execute a function within the context of every matched element.

jQuery

**size( )**

The number of elements in the jQuery object.

Num

**length**

The number of elements in the jQuery object.

Num

**selector**

A selector representing selector originally passed to jQuery().

String

**context**

The DOM node context originally passed to jQuery () (if none was passed then context will be equal to the document).

El

**eq( position )**

Reduce the set of matched elements to a single element.

jQuery

**get( )**

Access all matched DOM elements.

a<El>

**get( index )**

Access a single matched DOM element at a specified index in the matched set.

El

**index( subject )**

Searches every matched element for the object and returns the index of the element, if found, starting with zero. If a jQuery object is passed, only the first element is checked.

Num

★ CORE / 3. DATA

**data( name )**

Returns value at named data store for the element, as set by data(name, value).

Any

**data( name, value )**

Stores the value in the named spot.

jQuery

**removeData( name )**

Removes named data store from an element.

jQuery

**queue( name )**

Returns a reference to the first element's queue (which is an array of functions).

a<Fx>

**queue( name, callback )**

Adds a new function, to be executed, onto the end of the queue of all matched elements.

jQuery

**queue( name, queue )**

Replaces the queue of all matched element with this new queue (the array of functions).

jQuery

**dequeue( name )**

Removes a queued function from the front of the queue and executes it.

jQuery

★ CORE / 4. PLUGIN

**jQuery.fn.extend( object )**

Extends the jQuery element set to provide new methods (used to make a typical jQuery plugin).

jQuery

**jQuery.extend( object )**

Extends the jQuery object itself.

jQuery

★ CORE / 5. INTEROPERABILITY

**jQuery.noConflict( )**

Run this function to give control of the \$ variable back to whichever library first implemented it.

jQuery

**jQuery.noConflict( extreme )**

Extends the jQuery object itself. Use with discretion.

jQuery

★ ATTRIBUTES / 1. ATTR

**attr( name )**

Access a property on the first matched element. This method makes it easy to retrieve a property value from the first matched element.

If the element does not have an attribute with such a name, undefined is returned.

Object

T Attributes include title, alt, src, href, width, style, etc.

**attr( properties )**

Set a key/value object as properties to all matched elements.

jQuery

**attr( key, value )**

Set a single property to a value, on all matched elements.

jQuery

**attr( key, fn )**

Set a single property to a computed value, on all matched elements.

jQuery

**removeAttr( name )**

Remove an attribute from each of the matched elements.

jQuery

★ ATTRIBUTES / 2. CLASS

**addClass( class )**

Adds the specified class(es) to each of the set of matched elements.

jQuery

**hasClass( class )**

Returns true if the specified class is present on at least one of the set of matched elements.

0-1

**removeClass( class )**

Removes all or the specified class(es) from the set of matched elements.

jQuery

**toggleClass( class )**

Adds the specified class if it is not present, removes the specified class if it is present.

jQuery

**toggleClass( class, switch )**

Adds the specified class if the switch is true, removes the specified class if the switch is false.

jQuery

★ ATTRIBUTES / 3. HTML

**html( )**

Get the html contents (innerHTML) of the first matched element. This property is not available on XML documents (although it will work for XHTML documents).

T How to use:

```
$( "p" ).click( function () {
    var htmlStr = $( this ).html();
    $( this ).text( htmlStr );
} );
```

String

**html( val )**

Set the html contents of every matched element. This property is not available on XML documents (although it will work for XHTML documents).

jQuery

T How to use:

```
$( "div" ).html(
    <span class= 'red' >Hello
    <b>Again</b></span> );
```

★ ATTRIBUTES / 4. TEXT

**text( )**

Get the combined text contents of all matched elements.

String

T How to use:

```
var str = $( "p: first" ).text();
$( "p: last" ).html( str );
```

**text( val )**

Set the text contents of all matched elements.

jQuery

★ ATTRIBUTES / 5. VALUE

**val( )**

Get the input value of the first matched element.

T How to use:

```
$( "input" ).keyup( function () {
    var value = $( this ).val();
    $( "p" ).text( value );
} ).keyup();
```

String | a<>

**val( val )**

Set the text contents of all matched elements. Checks, or selects, all the radio buttons, checkboxes, and select options that match the set of values.

jQuery

★ TRAVERSING / 1. FILTERING

**eq( index )**

Reduce the set of matched elements to a single element.

jQuery

**T** How to use:

```
$("#div").eq(2).addClass("blue");
```

**filter( expr )**

Removes all elements from the set of matched elements that do not match the specified expression(s).

jQuery

**T** How to use:

```
$("#div").css("background", "#c8ebcc")
    .filter(".middle")
    .css("border-color", "red");
```

**filter( fn )**

Removes all elements from the set of matched elements that do not match the specified function.

jQuery

**is( expr )**

Checks the current selection against an expression and returns true, if at least one element of the selection fits the given expression.

0-1

**map( callback )**

Translate a set of elements in the jQuery object into another set of values in a jQuery array (which may, or may not contain elements).

jQuery

**not( expr )**

Removes elements matching the specified expression from the set of matched elements.

jQuery

**slice( start, end )**

Selects a subset of the matched elements.

jQuery

★ TRAVERSING / 2. CHAINING

**andSelf( )**

Add the previous selection to the current selection.

jQuery

**T** How to use:

```
$("#div").find("p").andSelf().addClass("border");
$("#div").find("p").addClass("background");
```

**end( )**

Revert the most recent 'destructive' operation, changing the set of matched elements to its previous state (right before the destructive operation).

jQuery

★ TRAVERSING / 3. FINDING

**add( expr )**

Adds more elements, matched by the given expression, to the set of matched elements.

jQuery

**children( expr )**

Get a set of elements containing all of the unique immediate children of each of the matched set of elements.

jQuery

**closest( expr )**

Get a set of elements containing the closest parent element that matches the specified selector, the starting element included.

jQuery

**contents( )**

Find all the child nodes inside the matched elements (including text nodes), or the content document, if the element is an iframe.

jQuery

**find( expr )**

Searches for descendent elements that match the specified expression.

jQuery

**next( expr )**

Get a set of elements containing the unique next siblings of each of the given set of elements.

jQuery

**nextAll( expr )**

Find all sibling elements after the current element.

jQuery

**offsetParent( )**

Returns a jQuery collection with the positioned parent of the first matched element.

jQuery

**parent( expr )**

Get the direct parent of an element. If called on a set of elements, parent returns a set of their unique direct parent elements.

jQuery

**parents( expr )**

Get a set of elements containing the unique ancestors of the matched set of elements (except for the root element).

jQuery

**prev( expr )**

Get a set of elements containing the unique previous siblings of each of the matched set of elements.

jQuery

**prevAll( expr )**

Find all sibling elements in front of the current element.

jQuery

**siblings( expr )**

Get a set of elements containing all of the unique siblings of each of the matched set of elements.

jQuery

★ MANIPULATION / 1. CONTENTS

**html( )**

Get the html contents (innerHTML) of the first matched element.

String

**T** This property is not available on XML documents

**html( val )**

Set the html contents of every matched element.

jQuery

**T** This property is not available on XML documents

**text( )**

Get the combined text contents of all matched elements.

String

**text( val )**

Set the text contents of all matched elements.

jQuery

★ MANIPULATION / 2. INSERTING INSIDE

**append( content )**

Append content to the inside of every matched element.

jQuery

**appendTo( selector )**

Append all of the matched elements to another, specified, set of elements.

jQuery

**prepend( content )**

Prepend content to the inside of every matched element.

jQuery

**prependTo( selector )**

Prepend all of the matched elements to another, specified, set of elements.

jQuery

★ MANIPULATION / 3. INSERTING OUTSIDE

**after( content )**

Insert content after each of the matched elements.

jQuery

**before( content )**

Insert content before each of the matched elements.

jQuery

**insertAfter( selector )**

Insert all of the matched elements after another, specified, set of elements.

jQuery

**insertBefore( selector )**

Insert all of the matched elements before another, specified, set of elements.

jQuery

★ MANIPULATION / 4. INSERTING AROUND

**wrap( html )**

Wrap each matched element with the specified HTML content.

jQuery

**wrap( elem )**

Wrap each matched element with the specified element.

jQuery

**wrapAll( html )**

Wrap all the elements in the matched set into a single wrapper element.

jQuery

**wrapAll( elem )**

Wrap all the elements in the matched set into a single wrapper element.

jQuery

**wrapInner( html )**

Wrap the inner child contents of each matched element (including text nodes) with an HTML structure.

jQuery

**wrapInner( elem )**

Wrap the inner child contents of each matched element (including text nodes) with a DOM element.

jQuery

★ MANIPULATION / 5. REPLACING

**replaceWith( content )**

Replaces all matched elements with the specified HTML or DOM elements. This returns the jQuery element that was just replaced, which has been removed from the DOM.

jQuery

**replaceAll( selector )**

Replaces the elements matched by the specified selector with the matched elements.

jQuery

★ MANIPULATION / 6. REMOVING

**empty( )**

Remove all child nodes from the set of matched elements.

jQuery

**remove( expr )**

Removes all matched elements from the DOM.

jQuery

★ MANIPULATION / 7. COPYING

**clone()**  
Clone matched DOM Elements and select the clones. jQuery

**clone(bool)**  
Clone matched DOM Elements, and all their event handlers, and select the clones. jQuery

★ CSS / 1. CSS

**css(name)**  
Return a style property on the first matched element. String

**css(properties)**  
Set a key/value object as style properties to all matched elements. jQuery

**css(name, value)**  
Set a single style property to a value on all matched elements. jQuery

★ CSS / 2. POSITIONING

**offset()**  
Get the current offset of the first matched element, in pixels, relative to the document. Object{top, left}

**offsetParent()**  
Returns a jQuery collection with the positioned parent of the first matched element. jQuery

**position()**  
Gets the top and left position of an element relative to its offset parent. Object{top, left}

**scrollTop()**  
Gets the scroll top offset of the first matched element. Int

**scrollTop(val)**  
When a value is passed in, the scroll top offset is set to that value on all matched elements. jQuery

**scrollLeft()**  
Gets the scroll left offset of the first matched element. Int

**scrollLeft(val)**  
When a value is passed in, the scroll left offset is set to that value on all matched elements. jQuery

★ CSS / 3. HEIGHT AND WIDTH

**height()**  
Get the current computed, pixel, height of the first matched element. Int

**height(val)**  
Set the CSS height of every matched element. jQuery

**width()**  
Get the current computed, pixel, width of the first matched element. Int

**width(val)**  
Set the CSS width of every matched element. jQuery

**innerHeight()**  
Gets the inner height (excludes the border and includes the padding) for the first matched element. Int

**innerWidth()**  
Gets the inner width (excludes the border and includes the padding) for the first matched element. Int

**outerHeight(margin)**  
Gets the outer height (includes the border and padding by default) for the first matched element. Int

**outerWidth(margin)**  
Get the outer width (includes the border and padding by default) for the first matched element. Int

★ EVENTS / 1. PAGE LOAD

**ready(fn)**  
Binds a function to be executed whenever the DOM is ready to be traversed and manipulated. jQuery

★ EVENTS / 2. LIVE EVENTS

**live(type, fn)**  
Binds a handler to an event (like click) for all current - and future - matched element. Can also bind custom events. jQuery

**die(type, fn)**  
This removes a bound live event. jQuery

★ EVENTS / 2. EVENT HANDLING

**bind(type, data, fn)**  
Binds a handler to one or more events (like click) for each matched element. Can also bind custom events. jQuery

**one(type, data, fn)**  
Binds a handler to one or more events to be executed once for each matched element. jQuery

**trigger(event, data)**  
Trigger an event on every matched element. jQuery

**triggerHandler(event, data)**  
Triggers all bound event handlers on an element (for a specific event type) WITHOUT executing the browser's default actions, bubbling, or live events. Object

**unbind(type, fn)**  
This does the opposite of bind, it removes bound events from each of the matched elements. jQuery

★ EVENTS / 3. INTERACTION HELPERS

**hover(over, out)**  
Simulates hovering (moving the mouse on, and off, an object). This is a custom method which provides an 'in' to a frequent task. jQuery

**toggle(fn, fn2, fn3, fn4, ...)**  
Apply all of the matched elements to another, specified, set of elements. jQuery

★ EVENTS / 4. EVENT HELPERS

**blur() / blur(fn)**  
Triggers / Bind a function to the blur event of each matched element. jQuery

**change() / change(fn)**  
Triggers / Bind a function to the change event of each matched element. jQuery

**click() / click(fn)**  
Triggers / Bind a function to the click event of each matched element. jQuery

**dblclick() / dblclick(fn)**  
Triggers / Bind a function to the dblclick event of each matched element. jQuery

**error() / error(fn)**  
Triggers / Bind a function to the error event of each matched element. jQuery

**focus() / focus(fn)**  
Triggers / Bind a function to the focus event of each matched element. jQuery

**keydown() / keydown(fn)**  
Triggers / Bind a function to the keydown event of each matched element. jQuery

**keypress() / keypress(fn)**  
Triggers / Bind a function to the keypress event of each matched element. jQuery

**keyup() / keyup(fn)**  
Triggers / Bind a function to the keyup event of each matched element. jQuery

**load(fn)**  
Binds a function to the load event of each matched element. jQuery

**mousedown(fn)**  
Binds a function to the mousedown event of each matched element. jQuery

**mouseenter(fn)**  
Bind a function to the mouseenter event of each matched element. jQuery

**mouseleave(fn)**  
Bind a function to the mouseleave event of each matched element. jQuery

**mousemove(fn)**  
Bind a function to the mousemove event of each matched element. jQuery

**mouseout(fn)**  
Bind a function to the mouseout event of each matched element. jQuery

**mouseover(fn)**  
Bind a function to the mouseover event of each matched element. jQuery

**mouseup(fn)**  
Bind a function to the mouseup event of each matched element. jQuery

**resize(fn)**  
Bind a function to the resize event of each matched element. jQuery

**scroll(fn)**  
Bind a function to the scroll event of each matched element. jQuery



**select( ) / select( fn )**

Triggers / Bind a function to the select event of each matched element. jQuery

**submit( ) / submit( fn )**

Triggers / Bind a function to the submit event of each matched element. jQuery

**unload( fn )**

Binds a function to the unload event of each matched element. jQuery

★ EFFECTS / 1. BASICS

**show( )**

Displays each of the set of matched elements if they are hidden. jQuery

**T** How to use:

```
$( "p" ).show()
```

**show( speed, callback )**

Show all matched elements using a graceful animation and firing an optional callback after completion. jQuery

**hide( )**

Hides each of the set of matched elements if they are shown. jQuery

**T** How to use:

```
$( "p" ).hide()
```

**hide( speed, callback )**

Hide all matched elements using a graceful animation and firing an optional callback after completion. 0-1

**toggle( )**

Toggle displaying each of the set of matched elements. jQuery

**T** How to use:

```
$( "button" ).click( function () {
    $( "p" ).toggle();
});
```

**toggle( switch )**

Toggle displaying each of the set of matched elements based upon the switch (true shows all elements, false hides all elements). jQuery

**toggle( speed, callback )**

Toggle displaying each of the set of matched elements using a graceful animation and firing an optional callback after completion. jQuery

★ EFFECTS / 2. SLIDING

**slideDown( speed, callback )**

Reveal all matched elements by adjusting their height and firing an optional callback after completion. jQuery

**slideUp( speed, callback )**

Hide all matched elements by adjusting their height and firing an optional callback after completion. jQuery

**slideToggle( speed, callback )**

Toggle the visibility of all matched elements by adjusting their height and firing an optional callback after completion. jQuery

★ EFFECTS / 3. FADING

**fadeIn( speed, callback )**

Fade in all matched elements by adjusting their opacity and firing an optional callback after completion. jQuery

**T** How to use:

```
$( document.body ).click( function () {
    $( "div: hidden: first" ).fadeIn( "slow" );
});
```

**fadeOut( speed, callback )**

Fade out all matched elements by adjusting their opacity to 0, then setting display to "none" and firing an optional callback after completion. jQuery

**fadeTo( speed, opacity, callback )**

Fade the opacity of all matched elements to a specified opacity and firing an optional callback after completion. jQuery

**T** How to use:

```
$( "p: first" ).click( function () {
    $( this ).fadeTo( "slow", 0.33 );
});
```

★ EFFECTS / 4. SETTINGS

**jQuery.fx.off**

Globally disable all animations. 0-1

**T** How to use:

```
jQuery.fx.off = true;
$( "input" ).click( function () {
    $( "div" ).toggle( "slow" );
});
```

★ EFFECTS / 5. CUSTOM

**animate( params, duration, easing, callback )**

A function for making custom animations. jQuery

**T** How to use:

```
$( "#go" ).click( function () {
    $( "#block" ).animate( {
        width: "70%",
        opacity: 0.4,
    }, 1500 );
});
```

**animate( params, options )**

A function for making custom animations. jQuery

**stop( clearQueue, gotoEnd )**

Stops all the currently running animations on all the specified elements. jQuery

★ AJAX / 1. AJAX REQUESTS

**jQuery.ajax( options )**

Load a remote page using an HTTP request. XMLHttpRequest

**load( url, data, callback )**

Load HTML from a remote file and inject it into the DOM. jQuery

**jQuery.get( url, data, callback, type )**

Load a remote page using an HTTP GET request. XMLHttpRequest

**jQuerygetJSON( url, data, callback, type )**

Load JSON data using an HTTP GET request. XMLHttpRequest

**jQuery.getScript( url, callback )**

Loads, and executes, a local JavaScript file using an HTTP GET request. XMLHttpRequest

**jQuery.post( url, data, callback, type )**

Load a remote page using an HTTP POST request. XMLHttpRequest

**T** How to use:

```
$.post( "test.php", { func:
    "getNameAndTime" },
function( data ) {
    alert( data.name ); // John
    console.log( data.time ); // 2pm
}, "json" );
```

★ AJAX / 2. EVENTS

**ajaxComplete( callback )**

Attach a function to be executed whenever an AJAX request completes. jQuery

**T** How to use:

```
$( "#msg" ).ajaxComplete( function
( event, request, settings ) {
    $( this ).append( "<li>Complete!</li>" );
});
```

**ajaxError( callback )**

Attach a function to be executed whenever an AJAX request fails. jQuery

**ajaxSend( callback )**

Attach a function to be executed before an AJAX request is sent. jQuery

**T** How to use:

```
$( "#msg" ).ajaxSend( function( evt, request,
settings ) {
    $( this ).append( "<li>Starting request at " +
settings.url + "</li>" );
});
```

**ajaxStart( callback )**

Attach a function to be executed whenever an AJAX request begins and there is none already active. jQuery

**ajaxStop( callback )**

Attach a function to be executed whenever all AJAX requests have ended. jQuery

**ajaxSuccess( callback )**

Attach a function to be executed whenever an AJAX request completes successfully. jQuery

★ AJAX / 3. MISC

**jQuery.ajaxSetup( options )**

Setup global settings for AJAX requests. String

**serialize( )**

Serializes a set of input elements into a string of data. String

**serializeArray( )**

Serializes all forms and form elements (like the .serialize() method) but returns a JSON data structure for you to work with. a<Object>

★ UTILITIES / 1. DETECTION

**jQuery.support** *Object*  
 A collection of properties that represent the presence of different browser features or bugs.

**jQuery.browser \*** *Map*  
 Contains flags for the useragent, read from navigator.userAgent. While it is unlikely jQuery.browser will be removed, every effort to use jQuery.support and proper feature detection should be made.

**jQuery.browser.version \*** *String*  
 The version number of the rendering engine for the user's browser.

**jQuery.boxModel \*** *0-1*  
 States if the current page, in the user's browser, is being rendered using the W3C CSS Box Model

★ UTILITIES / 2. ARRAYS AND OBJECTS

**jQuery.each( object, callback )** *Object*  
 A generic iterator function, which can be used to seamlessly iterate over both objects and arrays.

**jQuery.extend( deep, target, object1, objectN )** *Object*  
 Extend one object with one or more others, returning the modified object.

**jQuery.grep( array, callback, invert )** *a<>*  
 Finds the elements of an array which satisfy a filter function. The original array is not affected.

**jQuery.makeArray( obj )** *a<>*  
 Turns anything into a true array.

**jQuery.map( array, callback )** *a<>*  
 Translate all items in an array to another array of items.

**jQuery.inArray( value, array )** *Num*  
 Determine the index of the first parameter in the Array (-1 if not found).

**jQuery.merge( first, second )** *a<>*  
 Merge two arrays together.

**jQuery.unique( array )** *a<>*  
 Remove all duplicate elements from an array of elements. Note that this only works on arrays of DOM elements, not strings or numbers.

★ UTILITIES / 3. TEST OPERATIONS

**jQuery.isArray( obj )** *0-1*  
 Determine if the parameter passed is an array.  
**T** How to use:  
 \$("b").append( "" + **jQuery.isArray()** );

**jQuery.isFunction( obj )** *0-1*  
 Determine if the parameter passed is a Javascript function object.

★ UTILITIES / 4. STRING OPERATIONS

**jQuery.trim( str )** *String*  
 Remove the whitespace from the beginning and end of a string.  
**T** How to use:  
 \$("button").click(function () {  
     var str = " lots of spaces before and after ";  
     alert( "" + str + "" );  
  
     str = **jQuery.trim**(str);  
     alert( "" + str + " - no longer" );  
     });

★ UTILITIES / 5. URLS

**jQuery.param( obj )** *String*  
 Serializes an array of form elements or an object (core of .serialize() method).  
**T** How to use:  
 var params = { width:1680, height:1050 };  
 var str = **jQuery.param**(params);  
 \$("#results").text(str);

★ REFERENCE AND CREDITS

JQUERY 1.3 Visual Cheat Sheet

Designed © 2009 by Antonio Lupetti  
 • <http://woork.blogspot.com>  
 • <http://www.twitter.com/woork>  
 • <http://www.facebook.com/antoniolupetti>

Download or buy a printed copy of this Visual Cheat Sheet here:  
 • <http://woork.blogspot.com>

jQuery | The Write Less, Do More, JavaScript Library

jQuery is a fast and concise JavaScript Library that simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development. jQuery is designed to change the way that you write JavaScript.

jQuery Official Web Page • <http://jquery.com/>  
 Original documentation • <http://docs.jquery.com>  
 Tutorials • <http://docs.jquery.com/Tutorials>  
 Bug Traker • <http://dev.jquery.com/>  
 Discussion • <http://docs.jquery.com/Discussion>

jQuery is © of John Resig and the jQuery Team.  
 • <http://ejohn.org>  
 • <http://docs.jquery.com/Contributors>

THE \*  
 WORKING  
 BRAIN