jQuery 3 Cheat Sheet

Baljeet Rathi, Author

GOOD TO KNOW

Official documentation: http://api.jquery.com

Selectors

[name^=‘value’]	Select attributes starting with given value
[name$=‘value’]	Select attributes ending with given value
[name*=‘value’]	Select attributes containing given value as substring
[name|=‘value’]	Select attributes equal to given value, or given value followed by a hyphen
[name~=‘value’]	Select attributes with given value, delimited by spaces
:checked	Select all checked or selected elements
:focus	Select all focused elements
:disabled	Select all disabled elements
:hidden	Select all hidden elements
:visible	Select all visible elements
:input	Select *<input>*, *<textarea>*, *<select>* and *<button>* elements

By default selectors carry out their searches within the DOM starting at the document root, although you can offer an alternate context via an optional second parameter. Example below: searches for the *<span>* selector within the clicked element.

```
$(‘.alpha’).on(‘click’, function() {
  $(‘span’, this).addClass(‘beta’);
});
```

preventDefault() vs stopPropagation()

Stops event from bubbling up the DOM tree, preventing parent handlers from being notified

```
event.stopPropagation()
```

Prevents the default browser action (e.g. form submission), however the event will still bubble up.

```
event.preventDefault()
```

jQuery 3 Cheat Sheet

Baljeet Rathi, Author

Document Loading

Execute a function once the DOM has loaded:

```
$(function() {
  // DOM is ready here
});

$(document).ready(…) // deprecated as of jQuery 3
```

Execute function once element has been loaded:

```
$(‘selector’).on(‘load’, function() {
  // Element and sub-elements have completely loaded
});
```

Tree Traversal

.children()	Get children of each selected element optionally filtered by a selector
.find()	Get descendants of each selected element, optionally filtered by a selector
.next()	Get immediate-next sibling of each element, optionally filtered by selector
.prev()	Get immediate-previous sibling of each element, optionally filtered by selector
.parent()	Get parent of each selected element, optionally filtered by selector
.closest()	For each selected element, get the first one that matches a given selector by testing the element itself and traversing through its ancestors

Event Delegation

You can use event delegation to attach a single event listener to a parent element that will fire for all its descendants that match a given selector. This works regardless of whether those descendants exist now, or are added in the future.

```
$(‘.parent’).on(‘click’, ‘a’, function(event) {
  console.log(‘Delegation Successful’);
});
```

DOM Manipulation

.append()	Insert given content at the end of each selected element
.appendTo()	Insert selected elements at the end of given target elements
.clone()	Create a deep copy of a set of matched elements; boolean paramater indicates whether ev.handlers should be copied
.detach()	Remove matched elements from DOM but keep all associated jQuery data in case the elements are reinserted later
.remove()	Remove matched elements from DOM
.replaceWith()	Replace each selected element with new content and return the set of matched elements
.wrap()	Wrap given HTML around each selected element
.wrapInner()	Wrap given HTML around the content of each selected element

jQuery 3 Cheat Sheet

Baljeet Rathi, Author

Animation/Effects

.animate()	Perform animations on any numeric CSS property
.fadeToggle()	Display/hide elements by animating opacity
.slideToggle()	Display/hide elements by animating height
.show()	When no duration is offered, these methods will show and hide elements instantly; otherwise, they will do so by animating the width, height and opacity simultaneously

Utility Functions

data()	Store arbitrary data associated with specific elements
grep()	Find elements in an array which satisfy a filter; original array is left unchanged
trim()	Remove all new lines/spaces/tabs from the beginning and end of given string

Form Serialization

Serialization converts form values in a valid query string. *Note: a form elements' value will only be included in the serialized string if the element has a* ***name*** *attribute. Values from checkboxes and radio buttons are included only if they're checked.*

```
.serialize()
```

JSON

Load JSON encoded data from the server using a GET HTTP request: *https://goo.gl/J2rg2B*.

```
$.getJSON('path/to/data.json', function(data) {
  // do something with the JSON data
});
```

About the Author

Baljeet is a writer and web developer based in India. He's been involved with web development for 5+ years now.

Ajax

Make asynchronous HTTP requests to a server without refreshing the webpage. *Note: settings are passed as an object literal.*

```
$.ajax({
  type: 'POST', // type of request
  url: 'submit.php', // url to send request to
  cache: false, // browser cache requested webpages
  data: { 'value' : text }, // data to be sent to server
  success: function(r){
    console.log(r); // called if request succeeds
  },
  error: function(r){
     console.log(r); // called if request fails
  },
  complete: function(r){
    console.log(r); // called when request finishes
  },
});
```

You can also use a number of global event handlers: *https://goo.gl/qxqxHv*.