Cheatography

JQuery to JavaScript Quick Ref Guide Cheat Sheet

by dwapi via cheatography.com/43665/cs/13113/

DOM Selectors

$('#id')	document.getElementById('id')
$('tagName')	document.getElementsByTagName('tagName') document.querySelectorAll("tagName") //returns a NodeList
$('.className')	document.getElementsByClassName('className')
$('tagName.className')	document.querySelectorAll('tagName.className');
$('#id').text('text')	document.getElementById(id).innerHTML = "text
$('#id').attr('attr', 'new value')	document.getElementById(id).attribute = new value
.attr ^ or removeAttr()	document.getElementById("id").setAttribute("attr", "val"); // or getAttribute
$('#id').css({color:red})	Object.assign(document.getElementById('id).style,{fontSize:"20px",color: red});
$("[attribute='value']")	document.querySelectorAll('[attribute="value"]'); &npsp; **IE9+**

In POJS A NodeList object is a list (collection) of nodes extracted from a document

Events

$(#id).click(function(){})	document.getElementById("id").onclick =function(){}
$(#id).on('event', function(){})	element.addEventListener('click', function() { ... }, false);
$(#id).on('mouseover click', function () { })	['click', 'mouseover'].map(function(e) { window.addEventListener(e, mouseMoveHandler); });
$.ready(function (){ });	document.addEventListener("DOMContentLoaded", function(event) { }) **IE9+**
$(window).load(function(){ });	window.onload = doSomething or function(){ };
$('#image').load(function(){ });	imageObj.addEventListener('load', function() { }, false);

ready and DOMContentLoaded are executed as soon as the HTML has downloaded and the DOM is ready, but it won't wait for other assets.
$(window).load and window.onload are triggered after all external resources (like images) have finished loading

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

Enumerables		
$.merge([0, 1], [2, 3])	Joins two or more arrays, and returns a copy of the joined arrays	[0, 1].concat([2, 3])
$.each([52, 97], function(index, item){})	Calls a function for each array element	array.forEach(function(currentValue, index, arr), function(){})
$.grep([0, 1], function(item, index){})	Returns a new array with matching elements	[1,2].filter(function(currentValue, index, arr){})
$.map(array, function(item, index) { return doSomething });	Creates a new array with the result of calling a function on each array element	array.map(function(currentValue, index, arr){ return doSomething;})
$.unique(array)	Remove duplicates from an array	arr.filter(function(item, index, inputArray) { return inputArray.indexOf(item) == index; });
N/A	Reduce the values of an array to a single value (going left-to-right)	[1, 2, 3].reduce(function(total, num){ return total + num; })
N/A	Reduce the values of an array to a single value (going right-to-left)	[1, 2, 3].reduceRight(function(total, num){ return total + num; })
N/A	Removes the first element of an array, and returns that element	[1, 2, 3].shift()
N/A	Adds new elements to the beginning of an array, and returns the new length	[1, 2, 3].unshift(4, 5)
N/A	Add items to the array based on index (rather than shift or push)	[1, 2, 3, 4].splice(2, 0, "a", "b") // [1, 2, "a", "-b", 3, 4]

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

DOM Manipulation

var newElement = $(" ", {"class": "something"});	var newElement = document.createElement('div',{ class: 'something'});
$(newElement).text('text')	var node = document.createTextNode("This is a new paragraph."); newElement.appendChild(node);
$(#parentNode).append($(newElement))	var parentNode = document.getElementById("div1"); element.appendChild(para);
$(#parentNode).prepend($(newElement))	parentNode.insertBefore(newElement, parentNode.firstChild);
$(newElement).before($("#firstChild"));	parentNode.insertBefore(newElement, parentNode.firstChild);
$("#inner").wrap($(outer));	document.getElementById("outer").innerHTML = raw_html or newElement;
$(#parentNode).remove()	parentNode.removeChild(child);
$(#oldNode).replaceWith($(#newNode))	parentNode.replaceChild(newNode, oldNode);
$("#id").clone()	ocument.getElementById("id").cloneNode(true);
$("#wrap").is(":empty")	document.getElementById("id").hasChildNodes()
$(#id).addClass('newClass') or removeClass()	document.getElementById("myelement").classList.add("newClass"); or .remove() **IE10+**
$(#id).hasClass('myclass')	document.getElementById("myelement").classList.contains("myclass"); **IE110+**
$(#id).toggleClass('myclass')	document.getElementById("myelement").classList.toggle("myclass"); **IE110+**

toggleClass and toggle will remove a class if present, or add it if not

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com