Jquery 1.3.2 Cheat Sheet 2009-05-07 by Matt Kruse http://www.JavascriptToolbox.com/jquery/cheatsheet/ Twitter:@matt_kruse Home: http://mattkruse.com

Core

$(html) / $(element) / $(selector [,context])

$(func) === $(document).ready(func)

each (func)

```
$("img").each(function(i){
  this.src = "test" + i + ".jpg";
  return false; // stop looping over each
});
```

$()[i] === $().get(i)

$().length / $().size() / index(obj)

$().selector() / $().context()

Data

.data(key) / .data(key,val) / .removeData(key)	Store/retrieve/remove arbitrary data tied to elements
.queue(name) / .queue(name, [fn|queue])	Retrieve an element's queue or add to/replace existing queue

Attributes

attr(name) / attr(name,val) / attr({name:val})

attr(name, func)

removeAttr(name)

$("img").attr("title", function() { return this.src });

addClass(c) / hasClass(c) / removeClass(c) / toggleClass(c[,switch])

html() / html(content) / text() / text(content) / val() / val(value)

Traversing

add(expr) / add(html) / add(Element)	Append more elements to the set of matched elements
contains('text')	$("div").contains('text') === $("div :contains('text')")
filter(expr) / filter(func)	Leave elements matching expr or func returning true
find(expr)	$("p").find("span") === $("p span")
is(expr [,expr])	Returns true if any in set matches expr. Complex selectors ok
next([expr]) / prev(expr)	Only immediate next/prev sibling [if expr matches]
nextAll([expr]) / prevAll(expr)	All next/previous siblings [if expr matches]
not(expr) / not(Element)	Removes matched elements from list
parent([expr])	Immediate parent, if matches expr
parents([expr])	All parent elements matching expr
offsetParent()	Positioning offset parent
closest(expr) === parents(expr :first)	Find closest parent element matching expr
siblings([expr]) / children([expr])	andSelf() / end()

DOM Manipulation

before(content) / after(content)	Creates a new sibling before/after element
insertBefore(expr) / insertAfter(expr)	Attach selected elements as new sibling to others
prepend(content) / append(content)	Creates a new child node at the beginning/end
prependTo(expr) / appendTo(expr)	Attach selected elements to others, return attched
empty() - Removes all child nodes and content	remove()
wrap(html)	$("p").wrap(" ");
wrapAll(html|elem) / wrapInner(html|elem)	replaceWith(content) / replaceWith(expr)
clone([boolean])	Clone event handlers if passed true

CSS

css(name) - get val from first element in list only	css(key,val) / css({key:val, key:val})
offset() / position()	Returns {top,left} relative to doc or offset parent
scrollTop([num]) / scrollLeft([num])	Scroll position of first element only
height() / height(val) / width() / width(val)	innerHeight() / innerWidth()
outerHeight([bool]) / outerWidth([bool])	Pass false to ignore margins

Events (return false from handlers to cancel default action)

bind (type,data,func) / unbind (type,func)

```
function handler(event) {
  alert(event.data.foo);
}
$("p").bind("click", {foo: "bar"}, handler)
```

one (type, data, func) - execute only once	
hover (overfunc, outfunc)	
toggle (evenfunc, oddfunc)	
trigger (type, data)	Executes browser's default action also
Trigger an event: *event*()	Bind a function to an event: *event*(fn)

EventTypes: blur, change, click, dblclick, error, focus, keydown, keypress, keyup, load, mousedown, mouseenter, mouseleave, mousemove, mouseout, mouseover, mouseup, resize, scroll, select, submit, unload

trigger (type, data)	Executes browser's default action also
live (type,fn)	Handle event using event delegation
die (type,fn)	Remove events added w/ live().

Effects

Speeds: slow / normal / fast / #ms	toggle ([boolean]) / toggle (speed,fn)
animate (params, options)	animate (params, speed, easing, callback)
hide/show (speed [,callback])	stop (clearQue,gotoEnd)
fadeIn/fadeOut (speed, callback)	fadeTo (speed, opacity, callback)
slideDown/slideUp (speed, callback)	slideToggle (speed,callback)

Set jQuery.fx.off=true to disable all current and queued animations

Selectors

#id / .className	:visible / :hidden
:enabled / :disabled	:checked / :selected
:nth-child(n) / :first-child / :last-child / :only-child / :gt(n) / :lt(n)	
:first / :last	:eq(0) / :nth(0)
:header / :animated	:even / :odd
E:empty	has no children (including text nodes)
E:not(s)	does not match simple selector s
E F	F element descendant of an E element
E > F	F element child of an E element
E + F	F element immediately preceded by an E element
E ~ F	F element preceded by an E element
E[foo]	contains a "foo" attribute
E[foo=bar]	"foo" attribute value is exactly equal to "bar"
E[foo^=bar]	"foo" attribute value begins exactly with "bar"
E[foo$=bar]	"foo" attribute value ends exactly with "bar"
E[foo*=bar]	"foo" attribute value contains the substring "bar"
E[foo!=bar]	"foo" attribute is not equal to "bar"
E[foo~=bar]	space-delimited "foo" attribute contains "bar"
E[foo=bar][baz=bop]	Match multiple attributes
:parent	elements which have child elements (including text)
:contains('test')	elements which contain the specified text.
:input	All form elements, not just type=input

Types= :password :radio :checkbox :submit :image :reset :button :file

AJAX

$.ajax(properties)	async: true
$.ajaxSetup (properties)	beforeSend: func(xhr)
$.get (url, properties, fn(data))	cache: true (false=no caching)
$.getJSON (url,props,fn(json))	complete: func(xhr, textStatus)
$.getScript (url, callback)	contentType: String
$.post (url, props, fn(data))	data: {obj} | String
ajaxComplete (fn(xhr,props))	dataFilter: func(data,type) - return sanitized data
ajaxError (fn(xhr,props))	dataType: [xml,html,script,json,jsonp,text]
ajaxSend (fn(xhr,props))	error: func(xhr, textStatus, exception)
ajaxStart (fn(xhr,props))	global: true (fire global events)
ajaxStop (fn(xhr,props))	ifModified: false
ajaxSuccess (fn(xhr,props))	jsonp: String
.serialize()	processData:true
.serializeArray()	scriptCharset: String
.load (url, props, fn(responseText,status,xhr))	success: func(data, textStatus)
partial content using selector in url:	timeout: Number
$("#feed").load("feeds.php .results",	type: [POST,GET]
{limit: 25},	url: string
function(text,status,xhr) { alert("Loaded 25!");}	username: String / password: String (for auth)
);	xhr: func (to create the XMLHttpRequest object)

Feature/Browser Detection

$.support.*property*	Check for browser support of features:
boxModel, cssFloat, hrefNormalized, htmlSerialize, leadingWhitespace, noCloneEvent, objectAll, opacity, scriptEval, style, tbody	
$.browser.version	$.browser.[safari, opera, msie, mozilla]

Misc

$.each (obj, func)

```
$.each( [0,1,2], function(i, n){
  alert( "Item #" + i + ": " + n );
});
$.each( {name:"John",lang:"JS"},function(i,n){
  alert( "Name: " + i + ", Value: " + n );
});
```

$.trim (str) / $.unique (array)	
$.extend (target, prop1, propN)	
var options = { name: "bar" };	
$.extend({validate:false,name:'foo'}, options);	
Result == { validate: false, name: "bar" }	
$.map (array, func)	
$.map([0,1,2], function(n){ return n + 4; });	
$.grep (array, func, invert)	$.merge (array, array) - removes dupes
$.grep([0,1,2], function(n,i){ return n>0; }) == [1,2]	$.merge([0,1,2], [2,3,4]) === [0,1,2,3,4]
$.makeArray (obj) === [obj]	$.inArray (value, array)
$.isArray (obj) / $.isFunction (obj)	$.inArray('y',['x','y','z']) == 1 (-1 if not found)