

## Celestra v1.18.1 cheatsheet – <https://github.com/Serrin/Celestra/>

The "Celestra" and the "\_" objects contain these functions, except the "doc" (this is global) and the polyfills. Example: "\_ . qsa ( )"

| Basic API   | DOM  | Functional programming  |
|---|--|---|
| <code>doc = document</code><br><code>qsa(&lt;selector&gt;[,context]).each( fn (el,i) { el.arguments;} );</code><br><code>qs(&lt;selector&gt;[,context]).argument;</code><br><code>domReady(&lt;fn&gt;);</code><br><code>inherit(&lt;subclass&gt;,&lt;superclass&gt;);</code><br><code>random(&lt;max&gt;); or random(&lt;min&gt;,&lt;max&gt;);</code><br><code>getScript(&lt;url&gt;[,success]);</code><br><code>getScripts(&lt;scripts&gt;);</code><br><code>getStyle(&lt;href&gt;[,success]);</code><br><code>getStyles(&lt;styles&gt;);</code><br><code>getUrlVar([name]);</code><br><code>getUrlVarFromString(&lt;querrystr&gt;[,name]);</code><br><code>obj2string(&lt;object&gt;);</code><br><code>getType(&lt;variable&gt;[,type]);</code><br><code>extend([deep,]&lt;target&gt;,&lt;source1&gt;[,srcN]);</code><br><code>getFullscreen();</code><br><code>setFullscreenOn(&lt;selector&gt; or &lt;element&gt;);</code><br><code>setFullscreenOff();</code><br><code>getLocation(&lt;success&gt;[,error]);</code><br><code>getDoNotTrack();</code><br><code>form2array(&lt;form&gt;);</code><br><code>form2string(&lt;form&gt;);</code><br><code>constant(&lt;value&gt;);</code><br><code>identity(&lt;value&gt;);</code><br><code>noop();</code><br><code>repeat(&lt;iteration&gt;,&lt;callback&gt;);</code><br><code>noConflict();</code><br><code>version;</code> | <code>domCreate(&lt;type&gt;[,properties[,innerHTML]]);</code><br><code>domGetCSS(&lt;element&gt;,&lt;property&gt;);</code><br><code>domSetCSS(&lt;element&gt;,&lt;property&gt;,&lt;value&gt;);</code><br><code>domSetCSS(&lt;element&gt;,&lt;properties&gt;);</code><br><code>domFadeIn(&lt;element&gt;[, duration[,display]]);</code><br><code>domFadeOut(&lt;element&gt;[,duration]);</code><br><code>domFadeToggle(&lt;element&gt;[,duration[,display]]);</code><br><code>domShow(&lt;element&gt;[,display]);</code><br><code>domHide(&lt;element&gt;);</code><br><code>domToggle(&lt;element&gt;[,display]);</code><br><code>domOn(&lt;eventTarget&gt;,&lt;eventType&gt;,&lt;callback&gt;);</code><br><code>domOff(&lt;eventTarget&gt;,&lt;eventType&gt;,&lt;callback&gt;);</code><br><code>domTrigger(&lt;eventTarget&gt;,&lt;eventType&gt;);</code> | <code>toFunction(&lt;function&gt;);</code><br><code>bind(&lt;function&gt;,&lt;context&gt;);</code><br><code>forEach(&lt;collection&gt;,&lt;callback&gt;);</code><br><code>each(&lt;collection&gt;,&lt;callback&gt;);</code><br><code>map(&lt;collection&gt;,&lt;callback&gt;);</code><br><code>forIn(&lt;object&gt;,&lt;callback&gt;);</code><br><code>mapIn(&lt;object&gt;,&lt;callback&gt;);</code><br><code>toArray(&lt;object&gt;);</code><br><code>toObject(&lt;array&gt;);</code> |
| AJAX and CORS   |  |   |
|   | <code>getJSON(&lt;url&gt;,&lt;success&gt;);</code><br><code>getText(&lt;url&gt;,&lt;success&gt;);</code><br><code>getAjax(&lt;url&gt;,&lt;format&gt;,&lt;success&gt;[,error][,user&lt;,&amp;password&gt;]);</code><br><code>postAjax(&lt;url&gt;,&lt;data&gt;,&lt;format&gt;,&lt;success&gt;[,error][,user&lt;,&amp;password&gt;]);</code><br><code>getCors(&lt;url&gt;,&lt;format&gt;,&lt;success&gt;[,error][,user&lt;,&amp;password&gt;]);</code><br><code>postCors(&lt;url&gt;,&lt;data&gt;,&lt;format&gt;,&lt;success&gt;[,error][,user&lt;,&amp;password&gt;]);</code>   |   |
| Cookie  |  |   |
|   | <code>setCookie(&lt;name&gt;,&lt;value&gt;[,hours[,path[,domain[,secure[,HttpOnly]]]]]);</code><br><code>getCookie([name]);</code><br><code>hasCookie(&lt;name&gt;);</code><br><code>removeCookie(&lt;name&gt;[,path[,domain[,secure[,HttpOnly]]]]]);</code>   |   |
| Type checking   |  |   |
| <code>isString(&lt;value&gt;;, isChar(&lt;value&gt;;, isNumber(&lt;value&gt;;, isInteger(&lt;value&gt;;, isFloat(&lt;value&gt;;, isBoolean(&lt;value&gt;;, isObject(&lt;value&gt;; isEmptyObject(&lt;value&gt;;, isFunction(&lt;value&gt;;, isArray(&lt;value&gt;;, isEmptyArray(&lt;value&gt;;, isArraylike(&lt;value&gt;;, isNull(&lt;value&gt;;, isUndefined(&lt;value&gt;;, isNullOrUndefined(&lt;value&gt;;, isPrimitive(&lt;value&gt;;, isSymbol(&lt;value&gt;; ES6, isMap(&lt;value&gt;; ES6, isSet(&lt;value&gt;; ES6, isDate(&lt;value&gt;;, isRegExp(&lt;value&gt;;, isElement(&lt;value&gt;;</code>  |  |   |
| Polyfills   |  |   |
| <code>Array.from(), Array.of(), Object.create(), Object.assign(), ChildNode.after(), ChildNode.before(), ChildNode.remove(), ChildNode.replaceWith(), ParentNode.append(), ParentNode.prepend(), Array.prototype.includes(), String.prototype.includes(), NodeList.prototype.forEach(), Number.MIN_SAFE_INTEGER, Number.MAX_SAFE_INTEGER, Number.EPSILON, Number.isNaN(), isNaN(), Number.isInteger(), Number.isFinite(), Number.isSafeInteger()</code>   |  |   |