

## Celestra cheatsheet – v3.5.1 – <https://github.com/Serrin/Celestra/>

The celestra and/or the objects contain these functions, except the polyfills. Example: `_.qsa("p");`

Core API	DOM	Type checking
<code>inherit(&lt;subclass&gt;,&lt;superclass&gt;);</code> <code>randomInt([&lt;max&gt; or &lt;min&gt;,&lt;max&gt;);</code> <code>randomFloat([&lt;max&gt; or &lt;min&gt;,&lt;max&gt;);</code> <code>randomString([&lt;length&gt;,&lt;specChar&gt;]);</code> <code>b64Encode(&lt;string&gt;);</code> <code>b64Decode(&lt;string&gt;);</code> <code>javaHash(&lt;data&gt;[,&lt;hexa&gt;]);</code> <code>getUrlVar([&lt;name&gt;]);</code> <code>getUrlVarFromString(&lt;querystr&gt;[,&lt;name&gt;]);</code> <code>obj2string(&lt;object&gt;);</code> <code>getType(&lt;variable&gt;[,&lt;type&gt;]);</code> <code>extend([&lt;deep&gt;,&lt;target&gt;,&lt;source1&gt;[,&lt;srcN&gt;]);</code> <code>deepAssign(&lt;target&gt;,&lt;source1&gt;[,&lt;srcN&gt;]);</code> <code>hasOwn(&lt;object&gt;,&lt;property&gt;);</code> <code>forIn(&lt;object&gt;,&lt;callback&gt;);</code> <code>strRemoveTags(&lt;string&gt;);</code> <code>strReverse(&lt;string&gt;);</code> <code>strReplaceAll(&lt;str&gt;,&lt;search&gt;,&lt;replace&gt;);</code> <code>strCodePoints(&lt;string&gt;);</code> <code>strFromCodePoints(&lt;iter&gt;);</code> <code>strAt(&lt;string&gt;,&lt;pos&gt;);</code> <code>toFunction(&lt;function&gt;);</code> <code>bind(&lt;function&gt;,&lt;context&gt;);</code> <code>constant(&lt;value&gt;);</code> <code>identity(&lt;value&gt;);</code> <code>noop();</code> <code>T();</code> <code>F();</code> <code>noConflict();</code> <code>VERSION;</code>	<code>qsa(&lt;selector&gt;[,&lt;context&gt;]).forEach(&lt;fn&gt;);</code> <code>qs(&lt;selector&gt;[,&lt;context&gt;]).argument;</code> <code>domReady(&lt;function&gt;);</code> <code>domCreate(&lt;type&gt;[,&lt;properties&gt;,&lt;innerHTML&gt;]);</code> <code>domCreate(&lt;element descriptive object&gt;);</code> <code>domToElement(&lt;htmlString&gt;);</code> <code>domGetCSS(&lt;element&gt;,&lt;property&gt;);</code> <code>domSetCSS(&lt;element&gt;,&lt;property&gt;,&lt;value&gt;);</code> <code>domSetCSS(&lt;element&gt;,&lt;properties&gt;);</code> <code>domFadeIn(&lt;element&gt;[,&lt;duration&gt;,&lt;display&gt;]);</code> <code>domFadeOut(&lt;element&gt;[,&lt;duration&gt;]);</code> <code>domFadeToggle(&lt;elem.&gt;[,&lt;duration&gt;,&lt;display&gt;]);</code> <code>domShow(&lt;element&gt;[,&lt;display&gt;]);</code> <code>domHide(&lt;element&gt;);</code> <code>domToggle(&lt;element&gt;[,&lt;display&gt;]);</code> <code>domIsHidden(&lt;element&gt;);</code> <code>domSiblings(&lt;element&gt;);</code> <code>domGetCSSVar(&lt;name&gt;);</code> <code>domSetCSSVar(&lt;name&gt;,&lt;value&gt;);</code> <code>importScript(&lt;url&gt;[,&lt;success&gt;];</code> <code>importScripts(&lt;scripts&gt; or &lt;script1&gt;[,&lt;scN&gt;]);</code> <code>importStyle(&lt;href&gt;[,&lt;success&gt;];</code> <code>importStyles(&lt;styles&gt; or &lt;style1&gt;[,&lt;styleN&gt;]);</code> <code>setFullscreenOn(&lt;selector&gt; or &lt;element&gt;);</code> <code>setFullscreenOff();</code> <code>getFullscreen();</code> <code>form2array(&lt;form&gt;); and form2string(&lt;form&gt;);</code> <code>getDoNotTrack();</code> <code>getLocation(&lt;success&gt;[,&lt;error&gt;];</code> <code>createFile(&lt;filename&gt;,&lt;content&gt;[,&lt;dType&gt;];</code>	<code>isString(&lt;v&gt;); and isChar(&lt;v&gt;);</code> <code>isNumber(&lt;value&gt;);</code> <code>isNumeric(&lt;value&gt;);</code> <code>isFloat(&lt;value&gt;);</code> <code>isBigInt(&lt;value&gt;);</code> <code>isDate(&lt;value&gt;);</code> <code>isBoolean(&lt;value&gt;);</code> <code>isElement(&lt;value&gt;);</code> <code>isObject(&lt;value&gt;);</code> <code>isEmptyObject(&lt;value&gt;);</code> <code>isFunction(&lt;value&gt;);</code> <code>isArraylike(&lt;value&gt;);</code> <code>isSameArray(&lt;array1&gt;,&lt;array2&gt;);</code> <code>isEmptyArray(&lt;value&gt;);</code> <code>isTypedArray(&lt;value&gt;);</code> <code>isArrayBuffer(&lt;value&gt;);</code> <code>isNull(&lt;value&gt;);</code> <code>isUndefined(&lt;value&gt;);</code> <code>isNullOrUndefined(&lt;value&gt;);</code> <code>isNil(&lt;value&gt;);</code> <code>isPrimitive(&lt;value&gt;);</code> <code>isRegexp(&lt;value&gt;);</code> <code>isSymbol(&lt;value&gt;);</code> <code>isMap(&lt;value&gt;);</code> <code>isWeakMap(&lt;value&gt;);</code> <code>isSet(&lt;value&gt;);</code> <code>isWeakSet(&lt;value&gt;);</code> <code>isIterator(&lt;value&gt;);</code> <code>isIterable(&lt;value&gt;);</code> <code>isGenerator(&lt;value&gt;);</code>
AJAX and CORS		
<code>ajax(&lt;Options object&gt;);, getJson(&lt;url&gt;,&lt;success&gt;);, getText(&lt;url&gt;,&lt;success&gt;);</code>		
<b>Options object properties (* = default value):</b> url: string, data: string, queryType: <i>"ajax"/"cors"</i> , type: <i>"get"/"post"</i> , success: function, error: function, format: <i>"text"/"json"/"xml"</i> , user: string, password: string		
Cookie		
<code>setCookie(&lt;name&gt;,&lt;value&gt;[,&lt;hours&gt;,&lt;path&gt;,&lt;domain&gt;,&lt;secure&gt;,&lt;SameSite&gt;,&lt;HttpOnly&gt;]);, getCookie([&lt;name&gt;]);, hasCookie(&lt;name&gt;);, removeCookie(&lt;name&gt;[,&lt;path&gt;,&lt;domain&gt;,&lt;secure&gt;,&lt;SameSite&gt;,&lt;HOnly&gt;]);, clearCookies([&lt;path&gt;,&lt;domain&gt;,&lt;sec&gt;,&lt;SameSite&gt;,&lt;HOnly&gt;]);</code>		

Collections	Polyfills
<pre> isSuperset(&lt;superset&gt;,&lt;subset&gt;); arrayMerge([deep,]&lt;target&gt;,&lt;source1&gt;[,srcN]); zip(&lt;collection1&gt;[,collectionN]); and unzip(&lt;collection&gt;); uniqueArray(&lt;value&gt;); and uniquePush(&lt;array&gt;,&lt;value&gt;); arrayClear(&lt;array&gt;); and arrayRemove(&lt;array&gt;,&lt;value&gt;[,all]); min(&lt;collection&gt;); and minIndex(&lt;collection&gt;); max(&lt;collection&gt;); and maxIndex(&lt;collection&gt;); setUnion(&lt;collection1&gt;[,collectionN]); setIntersection(&lt;set1&gt;,&lt;set2&gt;); setDifference(&lt;set1&gt;,&lt;set2&gt;); setSymmetricDifference(&lt;set1&gt;,&lt;set2&gt;); arrayUnion(&lt;collection1&gt;[,collectionN]); arrayIntersection(&lt;collection1&gt;,&lt;collection2&gt;); arrayDifference(&lt;collection1&gt;,&lt;collection2&gt;); arraySymmetricDifference(&lt;collection1&gt;,&lt;collection2&gt;); arrayRange(&lt;start&gt;,&lt;end&gt;[,step]); and iterRange([start[,step[,end]]]); arrayCycle(&lt;collection&gt;[,n]); and iterCycle(&lt;iter&gt;[,n]); arrayRepeat(&lt;value&gt;[,n]);, and iterRepeat(&lt;value&gt;[,n]); sizeof(&lt;collection&gt;); item(&lt;collection&gt;,&lt;index&gt;); and itemOf(&lt;collection&gt;,&lt;index&gt;); forOf(&lt;collection&gt;,&lt;callback&gt;); and forEach(&lt;collection&gt;,&lt;callback&gt;); mapOf(&lt;collection&gt;,&lt;callback&gt;); and map(&lt;collection&gt;,&lt;callback&gt;); filterOf(&lt;collection&gt;,&lt;callback&gt;); hasOf(&lt;collection&gt;,&lt;value&gt;); findOf(&lt;collection&gt;,&lt;callback&gt;); everyOf(&lt;collection&gt;,&lt;callback&gt;); and someOf(&lt;collection&gt;,&lt;callback&gt;); noneOf(&lt;collection&gt;,&lt;callback&gt;); firstOf(&lt;collection&gt;); and lastOf(&lt;collection&gt;); sliceOf(&lt;collection&gt;[,begin[,end]]); reverseOf(&lt;collection&gt;); sortOf(&lt;collection&gt;); reduceOf(&lt;collection&gt;,&lt;callback&gt;[,initialvalue]); concatOf(&lt;collection1&gt;[,collectionN]); and flatOf(&lt;collection&gt;); enumerateOf(&lt;collection&gt;); joinOf(&lt;collection&gt;[,separator]); takeOf(&lt;collection&gt;[,n]); and takeWhile(&lt;collection&gt;,&lt;callback&gt;); takeRight(&lt;collection&gt;[,n]); and takeRightWhile(&lt;collection&gt;,&lt;callback&gt;); dropOf(&lt;collection&gt;[,n]); and dropWhile(&lt;collection&gt;,&lt;callback&gt;); dropRight(&lt;collection&gt;[,n]); and dropRightWhile(&lt;collection&gt;,&lt;callback&gt;); </pre>	<pre> Array.prototype.values(); Array.prototype.includes(); Array.prototype.flat(); Array.prototype.flatMap(); String.prototype.includes(); String.prototype.trimStart(); String.prototype.trimLeft(); String.prototype.trimEnd(); String.prototype.trimRight(); String.prototype.padStart(); String.prototype.padEnd(); String.prototype.repeat(); String.prototype.replaceAll(); String.prototype.matchAll(); String.prototype[Symbol.iterator](); Object.assign(); Object.fromEntries(); Object.entries(); Object.values(); Object.getOwnPropertyDescriptors(); RegExp.prototype.flags; NodeList.prototype.forEach(); ChildNode.after(); ChildNode.before(); ChildNode.remove(); ChildNode.replaceWith(); ParentNode.append(); ParentNode.prepend(); Element.prototype.matches(); Element.prototype.closest(); Element.prototype.toggleAttribute(); Element.prototype.getAttributeNames(); window.screenLeft; window.screenTop; globalThis; BigInt.prototype.toJSON(); GeneratorFunction(); </pre>