

window = Browser global object

PROPERTIES

- b**.closed check if window is closed
- n**.devicePixelRatio ratio vertical size pix
- b**.fullScreen check if window is fullscreen
- n**.innerWidth width size (incl. scrollbar)
- n**.innerHeight height size (incl. scrollbar)
- n**.outerWidth width size (incl. browser)
- n**.outerHeight height size (incl. browser)
- n**.length number of frames
- s**.name inner name of window
- s**.status bottom statusbar text

API/OBJECTS PROPERTIES

- o**.applicationCache offline resources API
- o**.console console browser API
- o**.crypto cryptographic API
- o**.history session page history API
- o**.location information about URL API
- o**.localStorage storage for site domain
- o**.sessionStorage storage until closed
- o**.navigator information about browser
- o**.performance data about performance

SCREEN PROPERTIES

- o**.screen information about screen
- n**.screenX horizontal pos browser/screen
- n**.screenY vertical pos browser/screen
- n**.pageXOffset horizontal pixels scrolled
- n**.pageYOffset vertical pixels scrolled

WINDOW PROPERTIES

- o**.opener window that opened this window
- o**.parent parent of current window/frame
- o**.self this window (equal to .window)
- o**.top top window of current win/frame

METHODS

- s**.btoa(str) encode string to base64
- s**.atob(str) decode base64 string to text
- z**.focus() request send window to front
- z**.blur() remove focus from window
- o**.getSelection(id) return Selection object
- z**.postMessage(msg, dst, transf) send
- o**.open(url, name, options) open popup
- z**.stop() stop window loading
- b**.find(str, case, back, wrap, word, fr, d)
- z**.print() open print document window

ANIMATION METHODS

- n**.requestAnimationFrame(cb(n))
- z**.cancelAnimationFrame(reqID)

TIMER METHODS

- n**.setTimeout(f(a...), ms, a...) delay&run
- z**.clearTimeout(id) remove timeout
- n**.setInterval(f(a...), ms, a...) run every
- z**.clearInterval(id) remove interval

SCREEN METHODS

- z**.scrollBy(x, y) scroll x,y pixels (relative)
- z**.scrollTo(x, y) scroll x,y pixels (absolute)
- z**.moveBy(x, y) move window by x,y (rel)
- z**.moveTo(x, y) move window to x,y (abs)
- z**.resizeBy(x, y) resize win by x,y (rel)
- z**.resizeTo(w, h) resize win to WxH (abs)

STYLESHEET METHODS

- o**.getComputedStyle(elem, pseudoelem)
- a**.matchMedia(mediaq) match CSSMQ

screen = info about screen / resolution

PROPERTIES

- n**.availTop top-from space available
- n**.availLeft left-from space available
- n**.availWidth width space available
- n**.availHeight height space available
- n**.width screen width resolution
- n**.height screen height resolution
- n**.colorDepth screen color depth (bits)
- n**.pixelDepth screen pixel depth (bits)

METHODS

- b**.lockOrientation(mode|modearray)
- b**.unlockOrientation() remove locks

console = unofficial console browser API

METHODS

- z**.assert(cond, str1|obj1...) set a assert
- z**.count(str) count (show number times)
- z**.dir(obj) show object (expanded debug)
- z**.group() open new message group
- z**.groupCollapsed() open new group coll.
- z**.groupEnd() close previous group
- z**.table(array|obj, colnames) show table
- z**.trace() show code trace
- z**.timeStamp(str) put time on timeline

PERFORMANCE METHODS

- z**.profile(name) start performance profile
- z**.profileEnd(name) stop perf. profile
- z**.time(name) start performance timer
- z**.timeEnd(name) stop perf. timer

LOG LEVEL METHODS

- z**.log(str1|obj1...) output message
- z**.info(str1|obj1...) output information
- z**.warn(str1|obj1...) output warning
- z**.error(str1|obj1...) output error

window = global interaction func.

METHODS

USER INTERACTION METHODS

- z**.alert(str) show message (ok button)
- s**.prompt(str, def) ask answer to user
- b**.confirm(str) show message (ok, cancel)

history = page history on tab

PROPERTIES

- n**.length number of pages in historytab
- n**.state return state top history stack

METHODS

- z**.back() go prev page (same as .go(-1))
- z**.forward() go next page (same as .go(1))
- z**.go(n) go n page (positive or negative)
- z**.pushState(obj, title, url) insert state
- z**.replaceState(obj, title, url) repl. state

storage localStorage / sessionStorage

PROPERTIES

- n**.length number of items in storage

METHODS

- s**.key(n) return key name on position n
- s**.getItem(key) return value of item key
- z**.setItem(key, value) set or update key
- z**.removeItem(key) delete item with key
- z**.clear() delete all items for current site

performance = info about performance

PROPERTIES

- o**.navigation info about redir/type nav.
- o**.timing info about latency-load perf.

METHODS

- n**.now() high precision timestamp

navigator = info about browser

PROPERTIES

- b**.cookieEnabled browser cookies on?
- n**.doNotTrack DNT privacy enabled?
- o**.geolocation user-info geolocation
- s**.language language in browser
- n**.maxTouchPoints max on device
- b**.onLine browser work in online mode?
- s**.userAgent identify browser of user

METHODS

- n**.vibrate(n|pattern) use device vibration

location = info about current URL

PROPERTIES

- s**.href full document url
- s**.protocol <https://www.emezeta.com/>
- s**.username <https://user:pass@www>
- s**.password <https://user:pass@www>
- s**.host <https://emezeta.com:81/>
- s**.hostname <https://emezeta.com:81/>
- s**.port <https://emezeta.com:81/>
- s**.pathname <http://emezeta.com/42/>
- s**.hash <http://emezeta.com/#contacto>
- s**.search <http://google.com/?q=emezeta>
- o**.searchParams search params object
- s**.origin source origin of document url

onClick="..." (HTML) .onclick = (JS func) 'click' (Listener)

e events (only popular events)

MOUSE EVENTS

- e**.onClick
- e**.onMouseDown
- e**.onMouseEnter
- e**.onMouseMove
- e**.onMouseOut
- e**.onDbIcClick
- e**.onMouseUp
- e**.onMouseLeave
- e**.onMouseOver
- e**.onWheel

KEYBOARD EVENTS

- e**.onKeyDown
- e**.onKeyPress
- e**.onKeyUp

LOAD/OBJECT EVENTS

- e**.onDOMContentLoaded
- e**.onAbort
- e**.onResize
- e**.onBeforeUnload
- e**.onLoad
- e**.onError
- e**.onScroll
- e**.onUnload

FORM/FIELDS EVENTS

- e**.onBlur
- e**.onChange
- e**.onInvalid
- e**.onReset
- e**.onFocus
- e**.onInput
- e**.onSelect
- e**.onSubmit

ANIMATION/TRANSITION EVENTS

- e**.onDragEnter
- e**.onDragStart
- e**.onDragOver
- e**.onDragLeave
- e**.onDragEnd
- e**.onDrag e.onDrop

ANIMATION/TRANSITION EVENTS

- e**.onAnimationStart
- e**.onAnimationIteration
- e**.onAnimationEnd
- e**.transitionEnd

d document = Document object

PROPERTIES

- s** .characterSet document charset
- s** .compatMode quirks or standard mode
- s** .cookie return all cookies doc string
- s** .designMode return design mode status
- s** .dir return direction text: "rtl" or "ltr"
- s** .doctype return document type (DTD)
- s** .domain return document domain
- s** .documentURI return document URL
- s** .lastModified return date/time modific.
- s** .origin return document's origin
- s** .readyState return current load status
- s** .referrer return previous page (referrer)
- s** .title return document title
- s** .URL return HTML document URL
- o** .location information about URL

ELEMENTS PROPERTIES

- o** .activeElement focused element
- o** .body return body element
- o** .currentScript return active script
- o** .defaultView return window element
- o** .documentElement first element (root)
- o** .head return head element
- o** .scrollingElement first scrollable elem.

DOCUMENT ARRAY PROPERTIES

- a** .anchors array of images elements
- a** .applets array of applets elements
- a** .embeds array of embeds elements
- a** .forms array of forms elements
- a** .images array of images elements
- a** .links array of links elements
- a** .plugins array of plugins elements
- a** .scripts array of scripts elements

STYLESHEET PROPERTIES

- a** .styleSheets array of style files elem
- o** .preferredStyleSheetSet preferred css
- o** .selectedStyleSheetSet selected css

METHODS

- o** .adoptNode(**node**) adopt from ext doc
- o** .createAttribute(**name**) create Attr obj
- o** .createDocumentFragment()
- o** .createElement(**tag**) create Element obj
- o** .createEvent(**type**) create Event object
- o** .createRange() create Range object
- o** .createTextNode(**text**) create TextNode
- o** .enableStyleSheetsForSet(**name**)
- o** .importNode(**node**, **desc**) import copy
- o** .getElementById(**id**) find elem with id
- a** .getElementsByName(**name**) w/ name
- o** .getSelection(**id**) return Selection object

r ClientRect() = Coords of element

PROPERTIES

- n** .top top coord of surrounding rect
- n** .right right coord of surrounding rect
- n** .bottom bottom coord of surrounding r.
- n** .left left coord of surrounding rect
- n** .width width coord of surrounding rect
- n** .height height coord of surrounding r.

e Element() = Element object

PROPERTIES

- s** .accessKey if exist, shortcut key
- o** .attributes array of Attr objects
- o** .classList DOMTokenList of classes
- s** .className classes list to string
- s** .id id string of element
- s** .name name string of element
- s** .tagName HTML tag of element

POSITION, SIZE AND SCROLL PROPERTIES

- n** .clientTop top border width element
- n** .clientLeft left border width element
- n** .clientWidth inner width element
- n** .clientHeight inner height element
- n** .scrollTop top-position in document
- n** .scrollLeft left-position in document
- n** .scrollWidth width of element
- n** .scrollHeight height of element

GET/SET HTML CODE PROPERTIES

- s** .innerHTML get/set HTML inside elem
- s** .outerHTML get/set HTML (incl. elem)

METHODS

- o** .closest(**selec**) closest ancestor
- a** .getElementsByClassName(**class**)
- a** .getElementsByTagName(**tag**)
- o** .querySelector(**selec**) return first elem
- a** .querySelectorAll(**selec**) return elems
- b** .matches(**selec**) match with this elem?
- .insertAdjacentHTML(**posstr**, **html**)

ATTRIBUTE METHODS

- b** .hasAttributes() exists attributes?
- b** .hasAttribute(**name**) exist attribute?
- s** .getAttribute(**name**) return value
- .removeAttribute(**name**) del attribute
- .setAttribute(**name**, **value**) set attrib.

CLIENTRECT (POSITION AND SIZES) METHODS

- o** .getBoundingClientRect() return pos.
- a** .getClientRects() return pos/size array

e Event() = Event on action

PROPERTIES

- b** .bubbles true=bubble, false=captures
- b** .cancelable event is cancelable?
- o** .currentTarget current element
- b** .defaultPrevented preventDefault() call
- n** .detail additional event info
- n** .eventPhase current stage (0-3)
- b** .isTrusted user action or dispatched
- o** .target reference to dispatched object
- n** .timeStamp time when was created
- s** .type type of event

METHODS

- .preventDefault() cancel event
- .stopImmediatePropagation()
- .stopPropagation() prevent being called

t EventTarget (use over elements)

METHODS

- .addEventListener(**ev**, **cb(ev)**, **capt**)
- .removeEventListener(**ev**, **cb(ev)**, **capt**)
- b** .dispatchEvent(**ev**)

a Attr() = Attribute object

PROPERTIES

- s** .name name of element attribute
- s** .value value of element attribute

t DOMTokenList() = List of classes

PROPERTIES

- n** .length number of items

METHODS

- b** .contains(**item**) check if item exists
- .add(**item**) add item to list
- s** .item(**n**) return item number **n**
- .remove(**item**) del item from list
- b** .toggle(**item**) del item if exist, add else

n Node() = Minor element (elem. or text)

PROPERTIES

- s** .baseURI absolute base URL of node
- s** .namespaceURI namespace of node
- s** .nodeName name of node
- s** .nodeType 1=element, 2=text, 9=doc
- s** .nodeValue value of node
- s** .prefix namespace prefix of node
- s** .textContent text of node and children

NAVIGATION PROPERTIES

- o** .childNodes children nodes collection
- o** .firstChild first children (include text)
- o** .lastChild last children (include text)
- o** .nextSibling immediate next node
- o** .previousSibling immediate prev node
- o** .parentElement immediate parent elem
- o** .parentNode immediate parent node
- o** .ownerDocument return document

METHODS

- o** .appendChild(**node**) add node to end
- o** .cloneNode(**child**) duplicate node
- o** .compareDocumentPosition(**node**)
- b** .contains(**node**) node is descendant?
- b** .hasChildNodes() node has childs?
- o** .insertBefore(**newnode**, **node**)
- b** .isDefaultNamespace(**nsURI**)
- b** .isEqualNode(**node**) check if are equal
- s** .lookupNamespaceURI() ret namesp.
- s** .lookupPrefix() return prefix for a ns
- .normalize() normalize-form children
- o** .removeChild(**node**) del node & return
- o** .replaceChild(**newnode**, **oldnode**)

c ChildNode()

METHODS

- o** .remove() remove specified node

p ParentNode()

PROPERTIES

- n** .childElementCount number of children
- o** .children children elements
- o** .firstElementChild first children elem.
- o** .lastElementChild last children elem.

n NonDocumentTypeChildNode()

PROPERTIES

- o** .nextElementSibling next element
- o** .previousElementSibling prev element