

## JavaScript

### *DOM Object*

#### Document Object Properties and Methods

Property/Method	Description
<b>document.adoptNode(node)</b>	Returns an adapted node from another document to this document.
<b>document.anchors</b>	Returns a collection of all the anchors in the document
<b>document.applets</b>	Returns a collection of all the applets in the document
<b>document.baseURI</b>	Returns the absolute base URI of a document
<b>document.body</b>	Returns the body element of the document
<b>document.close()</b>	Closes the output stream previously opened with document.open()
<b>document.cookie</b>	Returns all name/value pairs of cookies in the document
<b>document.createAttribute()</b>	Creates an attribute node
<b>document.createComment()</b>	Creates a Comment node with the specified text
<b>document.createDocumentFragment()</b>	Creates an empty DocumentFragment node
<b>document.createElement()</b>	Creates an Element node
<b>document.createTextNode()</b>	Creates a Text node
<b>document.doctype</b>	Returns the Document Type Declaration associated with the document
<b>document.documentElement</b>	Returns the Document Element of the document (the HTML element)
<b>document.documentMode</b>	Returns the mode used by the browser to render the document
<b>document.documentURI</b>	Sets or returns the location of the document
<b>document.domain</b>	Returns the domain name of the server that loaded the document
<b>document.domConfig</b>	Returns the configuration used when normalizeDocument() is invoked
<b>document.forms</b>	Returns a collection of all the forms in the document
<b>document.getElementById()</b>	Returns the element that has the ID attribute with the specified value
<b>document.getElementsByName()</b>	Accesses all elements with a specified name
<b>document.getElementsByTagName()</b>	Returns a NodeList containing all elements with the specified tagname
<b>document.images</b>	Returns a collection of all the images in the document
<b>document.implementation</b>	Returns the DOMImplementation object that handles this document
<b>document.importNode()</b>	Imports a node from another document
<b>document.inputEncoding</b>	Returns the encoding, character set, used for the document
<b>document.lastModified</b>	Returns the date and time the document was last modified
<b>document.links</b>	Returns a collection of all the links in the document
<b>document.normalize()</b>	Removes empty Text nodes, and joins adjacent nodes
<b>document.normalizeDocument()</b>	Removes empty Text nodes, and joins adjacent nodes
<b>document.open()</b>	Opens an HTML output stream to collect output from document.write()
<b>document.readyState</b>	Returns the (loading) status of the document
<b>document.referrer</b>	Returns the URL of the document that loaded the current document
<b>document.renameNode()</b>	Renames the specified node
<b>document.strictErrorChecking</b>	Sets or returns whether error-checking is enforced or not
<b>document.title</b>	Sets or returns the title of the document
<b>document.URL</b>	Returns the full URL of the document
<b>document.write()</b>	Writes HTML expressions or JavaScript code to a document
<b>document.writeln()</b>	Same as write(), but adds a newline character after each statement

## JavaScript

### *DOM Object*

Property/Method	Reasons for avoiding
<code>document.attributes</code>	Documents don't have attributes
<code>document.hasAttributes()</code>	Documents don't have attributes
<code>document.nextSibling</code>	Documents don't have siblings
<code>document.nodeName</code>	This is always #document
<code>document.nodeType</code>	This is always 9 (DOCUMENT_NODE)
<code>document.nodeValue</code>	Documents don't have an node value
<code>document.ownerDocument</code>	Documents don't have an owner document
<code>document.ownerElement</code>	Documents don't have an owner element
<code>document.parentNode</code>	Documents don't have a parent node
<code>document.previousSibling</code>	Documents don't have siblings
<code>document.textContent</code>	Documents don't have a text content

#### Element Object Properties and Methods

Property/Method	Description
<code>element.accessKey</code>	Sets or returns the accesskey for an element
<code>element.appendChild()</code>	Adds a new child node, to an element, as the last child node
<code>element.attributes</code>	Returns a NamedNodeMap of an element's attributes
<code>element.childNodes</code>	Returns a NodeList of child nodes for an element
<code>element.className</code>	Sets or returns the class attribute of an element
<code>element.clientHeight</code>	Returns the viewable height of an element
<code>element.clientWidth</code>	Returns the viewable width of an element
<code>element.cloneNode()</code>	Clones an element
<code>element.compareDocumentPosition()</code>	Compares the document position of two elements
<code>element.dir</code>	Sets or returns the text direction of an element
<code>element.firstChild</code>	Returns the first child of an element
<code>element.getAttribute()</code>	Returns the specified attribute value of an element node
<code>element.getAttributeNode()</code>	Returns the specified attribute node
<code>element.getElementsByTagName()</code>	Returns a collection of all child elements with the specified tagname
<code>element.getFeature()</code>	Returns an object which implements the APIs of a specified feature
<code>element.getUserData()</code>	Returns the object associated to a key on an element
<code>element.hasAttribute()</code>	Returns true if an element has the specified attribute, otherwise false
<code>element.hasAttributes()</code>	Returns true if an element has any attributes, otherwise false
<code>element.hasChildNodes()</code>	Returns true if an element has any child nodes, otherwise false
<code>element.id</code>	Sets or returns the id of an element
<code>element.innerHTML</code>	Sets or returns the content of an element
<code>element.insertBefore()</code>	Inserts a new child node before a specified, existing, child node
<code>element.isDefaultNamespace()</code>	Returns true if a specified namespaceURI is the default, otherwise false
<code>element.isEqualNode()</code>	Checks if two elements are equal
<code>element.isSameNode()</code>	Checks if two elements are the same node
<code>element.isSupported()</code>	Returns true if a specified feature is supported on the element
<code>element.lang</code>	Sets or returns the language code for an element
<code>element.lastChild</code>	Returns the last child of an element
<code>element.namespaceURI</code>	Returns the namespace URI of an element
<code>element.nextSibling</code>	Returns the next node at the same node tree level
<code>element.nodeName</code>	Returns the name of an element
<code>element.nodeType</code>	Returns the node type of an element
<code>element.nodeValue</code>	Sets or returns the value of an element
<code>element.normalize()</code>	Joins adjacent text nodes and removes empty text nodes in an element
<code>element.offsetHeight</code>	Returns the height of an element
<code>element.offsetWidth</code>	Returns the width of an element
<code>element.offsetLeft</code>	Returns the horizontal offset position of an element

## JavaScript

### *DOM Object*

#### Element Object Properties and Methods

Property/Method	Description
<code>element.offsetParent</code>	Returns the offset container of an element
<code>element.offsetTop</code>	Returns the vertical offset position of an element
<code>element.ownerDocument</code>	Returns the root element (document object) for an element
<code>element.parentNode</code>	Returns the parent node of an element
<code>element.previousSibling</code>	Returns the previous element at the same node tree level
<code>element.removeAttribute()</code>	Removes a specified attribute from an element
<code>element.removeAttributeNode()</code>	Removes a specified attribute node, and returns the removed node
<code>element.removeChild()</code>	Removes a child node from an element
<code>element.replaceChild()</code>	Replaces a child node in an element
<code>element.scrollHeight</code>	Returns the entire height of an element
<code>element.scrollLeft</code>	Returns the distance between the left edge of an element and the view
<code>element.scrollTop</code>	Returns the distance between the top edge of an element and the view
<code>element.scrollWidth</code>	Returns the entire width of an element
<code>element.setAttribute()</code>	Sets or changes the specified attribute, to the specified value
<code>element.setAttributeNode()</code>	Sets or changes the specified attribute node
<code>element.setAttributeNode()</code>	
<code>element.setAttributeNode()</code>	
<code>element.setUserData()</code>	Associates an object to a key on an element
<code>element.style</code>	Sets or returns the style attribute of an element
<code>element.tabIndex</code>	Sets or returns the tab order of an element
<code>element.tagName</code>	Returns the tag name of an element
<code>element.textContent</code>	Sets or returns the textual content of a node and its descendants
<code>element.title</code>	Sets or returns the title attribute of an element
<code>element.toString()</code>	Converts an element to a string
<code>odelist.item()</code>	Returns the node at the specified index in a NodeList
<code>odelist.length</code>	Returns the number of nodes in a NodeList

#### Attribute Object Properties and Methods

Property/Method	Description
<code>attr.isId</code>	Returns true if the attribute is of type Id, otherwise it returns false
<code>attr.name</code>	Returns the name of an attribute
<code>attr.value</code>	Sets or returns the value of the attribute
<code>attr.specified</code>	Returns true if the attribute has been specified, otherwise it returns false
<code>nodemap.getNamedItem()</code>	Returns a specified attribute node from a NamedNodeMap.
<code>nodemap.item()</code>	Returns the node at a specified index in a NamedNodeMap
<code>nodemap.length</code>	Returns the number of nodes in a NamedNodeMap
<code>nodemap.removeNamedItem()</code>	Removes a specified attribute node
<code>nodemap.setNamedItem()</code>	Sets the specified attribute node (by name)

## JavaScript

### *DOM Object*

Property/Method	Reasons for avoiding
<code>attr.appendChild()</code>	Attributes don't have child nodes
<code>attr.attributes</code>	Attributes don't have attributes
<code>attr.baseURI</code>	use <code>document.baseURI</code> instead
<code>attr.childNodes</code>	Attributes don't have child nodes
<code>attr.cloneNode()</code>	Get or set the <code>attr.value</code> instead
<code>attr.firstChild</code>	Attributes don't have child nodes
<code>attr.hasAttributes()</code>	Attributes don't have attributes
<code>attr.hasChildNodes</code>	Attributes don't have child nodes
<code>attr.insertBefore()</code>	Attributes don't have child nodes
<code>attr.isEqualNode()</code>	Makes no sense
<code>attr.isSameNode()</code>	Makes no sense
<code>attr.isSupported()</code>	Is always true
<code>attr.lastChild</code>	Attributes don't have child nodes
<code>attr.nextSibling</code>	Attributes don't have siblings
<code>attr.nodeName</code>	Use <code>attr.name</code> instead
<code>attr.nodeType</code>	This is always 2 (ATTRIBUTE_NODE)
<code>attr.nodeValue</code>	Use <code>attr.value</code> instead
<code>attr.normalize()</code>	Attributes cannot be normalized
<code>attr.ownerDocument</code>	This is always your HTML document
<code>attr.ownerElement</code>	This is the HTML element you used to access the attribute
<code>attr.parentNode</code>	This is the HTML element you used to access the attribute
<code>attr.previousSibling</code>	Attributes don't have siblings
<code>attr.removeChild</code>	Attributes don't have child nodes
<code>attr.replaceChild</code>	Attributes don't have child nodes
<code>attr.textContent</code>	Use <code>attr.value</code> instead

#### Mouse Events

Property	Description
<code>onclick</code>	The event occurs when the user clicks on an element
<code>ondblclick</code>	The event occurs when the user double-clicks on an element
<code>onmousedown</code>	The event occurs when a user presses a mouse button over an element
<code>onmousemove</code>	The event occurs when the pointer is moving while it is over an element
<code>onmouseover</code>	The event occurs when the pointer is moved onto an element
<code>onmouseout</code>	The event occurs when a user moves the mouse pointer out of an element
<code>onmouseup</code>	The event occurs when a user releases a mouse button over an element

#### Keyboard Events

Attribute	Description
<code>onkeydown</code>	The event occurs when the user is pressing a key
<code>onkeypress</code>	The event occurs when the user presses a key
<code>onkeyup</code>	The event occurs when the user releases a key

#### Frame/Object Events

Attribute	Description
<code>onabort</code>	The event occurs when an image is stopped from loading before completely loaded (for <code>&lt;object&gt;</code> )
<code>onerror</code>	The event occurs when an image does not load properly (for <code>&lt;object&gt;</code> , <code>&lt;body&gt;</code> and <code>&lt;frameset&gt;</code> )
<code>onload</code>	The event occurs when a document, frameset, or <code>&lt;object&gt;</code> has been loaded
<code>onresize</code>	The event occurs when a document view is resized
<code>onscroll</code>	The event occurs when a document view is scrolled
<code>onunload</code>	The event occurs once a page has unloaded (for <code>&lt;body&gt;</code> and <code>&lt;frameset&gt;</code> )

## JavaScript

### *DOM Object*

#### Form Events

Attribute	Description
<b>onblur</b>	The event occurs when a form element loses focus
<b>onchange</b>	The event occurs when the content of a form element, the selection, or the checked state have changed (for <input>, <select>, and <textarea>)
<b>onfocus</b>	The event occurs when an element gets focus (for <label>, <input>, <select>, <textarea>, and <button>)
<b>onreset</b>	The event occurs when a form is reset
<b>onselect</b>	The event occurs when a user selects some text (for <input> and <textarea>)
<b>onsubmit</b>	The event occurs when a form is submitted

#### Event Object Constants

Constant	Description
<b>CAPTURING_PHASE</b>	The current event phase is the capture phase (3)
<b>AT_TARGET</b>	The current event is in the target phase, i.e. it is being evaluated at the event target (1)
<b>BUBBLING_PHASE</b>	The current event phase is the bubbling phase (2)

#### Event Object Properties

Property	Description
<b>bubbles</b>	Returns whether or not an event is a bubbling event
<b>cancelable</b>	Returns whether or not an event can have its default action prevented
<b>currentTarget</b>	Returns the element whose event listeners triggered the event
<b>eventPhase</b>	Returns which phase of the event flow is currently being evaluated
<b>target</b>	Returns the element that triggered the event
<b>timeStamp</b>	Returns the time (in milliseconds relative to the epoch) at which the event was created
<b>type</b>	Returns the name of the event

#### Event Object Methods

Method	Description
<b>initEvent()</b>	Specifies the event type, whether or not the event can bubble, whether or not the event's default action can be prevented
<b>preventDefault()</b>	To cancel the event if it is cancelable, meaning that any default action normally taken by the implementation as a result of the event will not occur
<b>stopPropagation()</b>	To prevent further propagation of an event during event flow

#### EventTarget Object Methods

Method	Description
<b>addEventListener()</b>	Allows the registration of event listeners on the event target (IE8 = attachEvent())
<b>dispatchEvent()</b>	Allows to send the event to the subscribed event listeners (IE8 = fireEvent())
<b>removeEventListener()</b>	Allows the removal of event listeners on the event target (IE8 = detachEvent())

#### EventListener Object Methods

Method	Description
<b>handleEvent()</b>	Called whenever an event occurs of the event type for which the EventListener interface was registered

#### DocumentEvent Object Methods

Method	Description
<b>createEvent()</b>	

## JavaScript

### *DOM Object*

#### MouseEvent/Keyboard Event Object Properties

Property	Description
altKey	Returns whether or not the "ALT" key was pressed when an event was triggered
button	Returns which mouse button was clicked when an event was triggered
clientX	Returns the horizontal coordinate of the mouse pointer, relative to the current window, when an event was triggered
clientY	Returns the vertical coordinate of the mouse pointer, relative to the current window, when an event was triggered
ctrlKey	Returns whether or not the "CTRL" key was pressed when an event was triggered
keyIdentifier	Returns the identifier of a key
keyLocation	Returns the location of the key on the device
metaKey	Returns whether or not the "meta" key was pressed when an event was triggered
relatedTarget	Returns the element related to the element that triggered the event
screenX	Returns the horizontal coordinate of the mouse pointer, relative to the screen, when an event was triggered
screenY	Returns the vertical coordinate of the mouse pointer, relative to the screen, when an event was triggered
shiftKey	Returns whether or not the "SHIFT" key was pressed when an event was triggered

#### MouseEvent/Keyboard Event Object Methods

Method	Description
initMouseEvent()	Initializes the value of a MouseEvent object
initKeyboardEvent()	Initializes the value of a KeyboardEvent object