



► [Table of contents](#)

Window.location

The `Window.location` read-only property returns a [Location](#) object with information about the current location of the document.

Though `Window.location` is a *read-only* `Location` object, you can also assign a [DOMString](#) to it. This means that you can work with `location` as if it were a string in most cases: `location = 'http://www.example.com'` is a synonym of `location.href = 'http://www.example.com'`.

See [Location](#) for all available properties.

Syntax

```
var oldLocation = location;  
location = newLocation;
```




Examples

Basic Example

```
alert(location); // alerts "https://developer.mozilla.org/en-US/docs/Web/API/Window.location" 
```

Example #1: Navigate to a new page

Whenever a new value is assigned to the location object, a document will be loaded using the URL as if `location.assign()` had been called with the modified URL.

Note that [navigation-related sandbox flags](#)  may result in an exception being thrown and the navigation failing.

```
location.assign("http://www.mozilla.org"); // or  
location = "http://www.mozilla.org";
```



Example #2: Reloading the current page

```
location.reload();
```



Example #3

Consider the following example, which will reload the page by using the [replace\(\)](#) method to insert the value of `location.pathname` into the hash:

```
function reloadPageWithHash() {  
    var initialPage = location.pathname;  
    location.replace('http://example.com/#' + initialPage);  
}
```



Example #4: Display the properties of the current URL in an alert dialog:

```
function showLoc() {  
    var oLocation = location, aLog = ["Property (Typeof): Value", "location"];  
    for (var sProp in oLocation){  
        aLog.push(sProp + " (" + (typeof oLocation[sProp]) + "): " + (oLocation[sProp]));  
    }  
    alert(aLog.join("\n"));  
}  
  
// in html: <button onclick="showLoc();">Show location properties</button>
```



Example #5: Send a string of data to the server by modifying the `search` property:

```
function sendData (sData) {
    location.search = sData;
}

// in html: <button onclick="sendData('Some data');">Send data</button>
```

The current URL with "?Some%20data" appended is sent to the server (if no action is taken by the server, the current document is reloaded with the modified search string).

Example #6: Using bookmarks without changing the **hash** property:

```
<!doctype html>
<html>
<head>
<meta charset="UTF-8"/>
<title>MDN Example</title>
<script>
function showNode (oNode) {
    document.documentElement.scrollTop = oNode.offsetTop;
    document.documentElement.scrollLeft = oNode.offsetLeft;
}

function showBookmark (sBookmark, bUseHash) {
    if (arguments.length === 1 || bUseHash) { location.hash = sBookmark; }
    var oBookmark = document.querySelector(sBookmark);
    if (oBookmark) { showNode(oBookmark); }
}
</script>
<style>
span.intLink {
    cursor: pointer;
    color: #0000ff;
    text-decoration: underline;
}
</style>

</head>

<body>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam
<p>Duis lobortis sapien quis nisl luctus porttitor. In tempor semper
```

```

<p id="myBookmark1">[&nbsp;<span class="intLink" onclick="showBookm
<p>Vivamus blandit massa ut metus mattis in fringilla lectus imperc
<p>Nam faucibus, ligula eu fringilla pulvinar, lectus tellus iaculi
<p>Aenean viverra varius mauris, sed elementum lacus interdum non.
<p>Suspendisse turpis nisl, consectetur in lacinia ut, ornare vel n
<p>Curabitur est ipsum, porta ac viverra faucibus, eleifend sed ero
<p>Vestibulum dignissim erat vitae lectus auctor ac bibendum eros s
<p id="myBookmark2">[&nbsp;<span class="intLink" onclick="showBookm
<p>Phasellus tempus fringilla nunc, eget sagittis orci molestie vel
<p>Fusce cursus pulvinar aliquam. Duis justo enim, ornare vitae ele
<p id="myBookmark3"><em>Here is the bookmark #3</em></p>
<p>Proin vitae sem non lorem pellentesque molestie. Nam tempus mass
<p>Fusce metus velit, pharetra at vestibulum nec, facilisis porttiti
<p>Fusce aliquet molestie dolor, in ornare dui sodales nec. In mole
<p>Etiam in blandit tellus. Integer sed varius quam. Vestibulum dap
<p>Etiam placerat dui ut sem ornare vel vestibulum augue mattis. Se
<p>Mauris vel odio vel nulla facilisis lacinia. Aliquam ultrices es
<p>Vestibulum id neque nec turpis iaculis pulvinar et a massa. Vest
<p>Pellentesque a lorem nulla, in tempor justo. Duis odio nisl, dig
<p>Sed id nulla mi, eget suscipit eros. Aliquam tempus molestie rut
<p>Nullam commodo suscipit lacus non aliquet. Phasellus ac nisl lor
</body>
</html>

```

...the same thing but with an animated page scroll:

```

var showBookmark = (function () {
    var _useHash, _scrollX, _scrollY, _nodeX, _nodeY, _itFrame, _scr
    /*
     * nDuration: the duration in milliseconds of each frame
     * nFrames: number of frames for each scroll
     */
    nDuration = 200, nFrames = 10;

    function _next () {
        if (_itFrame > nFrames) { clearInterval(_scrollId); _scrollId =
        _isBot = true;
        document.documentElement.scrollTop = Math.round(_scrollY + (_node
        document.documentElement.scrollLeft = Math.round(_scrollX + (_noc
        if (_useHash && _itFrame === nFrames) { location.hash = _bookMark
        _itFrame++;
    }

    function _chkOwner () {

```

```

if (!_isBot) { _isBot = false; return; }
if (_scrollId > -1) { clearInterval(_scrollId); _scrollId = -1; }
}

if (window.addEventListener) { window.addEventListener("scroll",
else if (window.attachEvent) { window.attachEvent("onscroll", _ch

return function (sBookmark, bUseHash) {
    var oNode = document.querySelector(sBookmark);
    _scrollY = document.documentElement.scrollTop;
    _scrollX = document.documentElement.scrollLeft;
    _bookMark = sBookmark;
    _useHash = bUseHash === true;
    _nodeX = oNode.offsetLeft;
    _nodeY = oNode.offsetTop;
    _itFrame = 1;
    if (_scrollId === -1) { _scrollId = setInterval(_next, Math.round
    };
})();

```

Specifications

Specification

[HTML Standard \(HTML\)](#)

[# the-location-interface](#)

Browser compatibility

[Report problems with this compatibility data on GitHub](#) 

location	
Chrome	1
Edge	12
Firefox	1 ★
Internet Explorer	4
Opera	3

Safari	1
WebView Android	1
Chrome Android	18
Firefox for Android	4 ★
Opera Android	10.1
Safari on iOS	1
Samsung Internet	1.0
window.location.origin	
Chrome	31
Edge	12
Firefox	21
Internet Explorer	11
Opera	?
Safari	7
WebView Android	37
Chrome Android	31
Firefox for Android	21
Opera Android	?
Safari on iOS	7
Samsung Internet	2.0



Full support



Compatibility unknown



See implementation notes.

See also

- The interface of the returned value, [Location](#).
- A similar information, but attached to the document, [Document.location](#).
- [Manipulating the browser history](#).
- [hashchange](#)

Found a problem with this page?

- [Source on GitHub](#)
- [Report a problem with this content on GitHub](#)
- Want to fix the problem yourself? See [our Contribution guide](#).

Last modified: Jun 17, 2021, [by MDN contributors](#)

Change your language

English (US) ▼

Change language

Related Topics

[Window](#)

▼ Properties

 [applicationCache](#)

 [caches](#)

[closed](#)

[console](#)

 [controllers](#)

 crossOriginIsolated

crypto

customElements

 defaultStatus

devicePixelRatio

 dialogArguments

 directories

document

event

frameElement

frames

 fullScreen

history

indexedDB

innerHeight

innerWidth

isSecureContext

isSecureContext

length

localStorage

location

locationbar

menubar

 mozAnimationStartTime

mozInnerScreenX

mozInnerScreenY

  mozPaintCount

name

navigator

 onabort

onafterprint

onanimationcancel

onanimationend

 onanimationiteration

 onappinstalled

 onauxclick

onbeforeinstallprompt

onbeforeprint

onbeforeunload

onblur

oncancel

oncanplay

oncanplaythrough

onchange

onclick

 onclose

oncontextmenu

oncuechange

ondblclick

ondevicemotion

ondeviceorientation

ondeviceorientationabsolute

 ondragdrop

ondurationchange

onended

onerror

onfocus

 onformdata

 ongamepadconnected

 ongamepaddisconnected

ongotpointercapture

onhashchange

oninput

oninvalid

onkeydown

 onkeypress

onkeyup

 onlanguagechange

onload

onloadeddata

onloadedmetadata

onloadend

onloadstart

onlostpointercapture

onmessage

onmessageerror

onmousedown

onmouseenter



onmouseleave

onmousemove

onmouseout

onmouseover

onmouseup

  onmozbeforepaint

 onpaint

onpause

onplay

onplaying

onpointercancel

onpointerdown

onpointerenter

onpointerleave

onpointermove

onpointerout

onpointerover

onpointerup

onpopstate

onrejectionhandled

onreset

onresize

onscroll

onselect

 onselectionchange

 onselectstart

onstorage

onsubmit

 ontouchcancel

 ontouchstart

ontransitioncancel

ontransitionend

onunhandledrejection

onunload

 onvrdisplayactivate

 onvrdisplayblur

 onvrdisplayconnect

 onvrdisplaydeactivate

 onvrdisplaydisconnect

 onvrdisplayfocus

 onvrdisplaypointerrestricted

 onvrdisplaypointerunrestricted

  onvrdisplaypresentchange

onwheel

opener

origin

outerHeight

outerWidth

pageXOffset

pageYOffset

parent

performance

personalbar

 pkcs11

screen

screenLeft

screenTop

screenX

screenY

scrollbars

 scrollMaxX

 scrollMaxY

scrollX

scrollY

self

sessionStorage

 sidebar

 speechSynthesis

status

statusbar

toolbar

top

 visualViewport

window

▼ Methods

alert()

atob()

  back()

blur()

btoa()

 cancelAnimationFrame()

cancelIdleCallback()

 captureEvents()

clearImmediate()

`clearInterval()`

`clearTimeout()`

`close()`

`confirm()`

 `convertPointFromNodeToPage()`

 `convertPointFromPageToNode`

`createImageBitmap()`

 `dump()`

 `fetch()`

 `find()`

`focus()`

  `forward()`

`getComputedStyle()`

`getDefaultComputedStyle()`

`getSelection()`

 `home()`

`matchMedia()`

`minimize()`

`moveBy()`

`moveTo()`

`open()`

 `openDialog()`

`postMessage()`

`print()`

`prompt()`

`queueMicrotask()`

 `releaseEvents()`

requestAnimationFrame()

  requestFileSystem()

requestIdleCallback()

resizeBy()

resizeTo()

 routeEvent()

scroll()

scrollBy()

 scrollByLines()

 scrollByPages()

scrollTo()

 setCursor()

 setImmediate()

setInterval()

setTimeout()

showDirectoryPicker()

 showModalDialog()

showOpenFilePicker()

showSaveFilePicker()

sizeToContent()

stop()

updateCommands()

▼ Events

event

afterprint

animationcancel

animationend

animationiteration

beforeprint

beforeunload

blur

copy

cut

DOMContentLoaded

error

focus

hashchange

 languagechange

load

message

messageerror

offline

online

orientationchange

pagehide

pageshow

paste

popstate

rejectionhandled

storage

transitioncancel

unhandledrejection

unload

vrdisplayconnect

[vrdisplaydisconnect](#)

[vrdisplaypresentchange](#)



[Web Technologies](#)

[Learn Web Development](#)

[About MDN](#)

[Feedback](#)

[About](#)

[MDN Web Docs Store](#)

[Contact Us](#)

[Firefox](#)

[MDN](#)  

[Mozilla](#)  

© 2005-2021 Mozilla and individual contributors. Content is available under these licenses.

[Terms](#) [Privacy](#) [Cookies](#)