MDN Web Docs
moz://a

<iframe>: The Inline Frame element

The `<iframe>` HTML element represents a nested browsing context, embedding another HTML page into the current one.

HTML Demo: <iframe> **Reset**

HTML | CSS

```
<iframe
id="inlineFrameExample"
    title="Inline Frame
Example"
    width="300"
    height="200"

 src="https://www.openstreetmap.org/export/embed.html?bbox=-0.004017949104309083%2C51.47612752641776%2C0.00030577182769775396%2C51.478569861898606&layer=mapnik">
</iframe>
```

Output

Each embedded browsing context has its own session history and document. The browsing context that embeds the others is called

the *parent browsing context*. The *topmost* browsing context — the one with no parent — is usually the browser window, represented by the `Window` object.

> **Warning:** Because each browsing context is a complete document environment, every `<iframe>` in a page requires increased memory and other computing resources. While theoretically you can use as many `<iframe>`s as you like, check for performance problems.

Content categories	Flow content, phrasing content, embedded content, interactive content, palpable content.
Permitted content	None.
Tag omission	None, both the starting and ending tag are mandatory.
Permitted parents	Any element that accepts embedded content.
Implicit ARIA role	No corresponding role
Permitted ARIA roles	`application`, `document`, `img`, `none`, `presentation`
DOM interface	`HTMLIFrameElement`

Attributes

This element includes the global attributes.

`allow`

Specifies a feature policy for the `<iframe>`. The policy defines what features are available to the `<iframe>` based on the origin of the request (e.g. access to the microphone, camera, battery, web-share API, etc.).

For more information and examples see: Using Feature Policy > The iframe allow attribute.

`allowfullscreen`

Set to `true` if the `<iframe>` can activate fullscreen mode by calling the `requestFullscreen()` method.

> **Note:** This attribute is considered a legacy attribute and redefined as `allow="fullscreen"`.

`allowpaymentrequest`

Set to `true` if a cross-origin `<iframe>` should be allowed to invoke the Payment Request API.

> **Note:** This attribute is considered a legacy attribute and redefined as `allow="payment"`.

`csp`

A Content Security Policy enforced for the embedded resource. See `HTMLIFrameElement.csp` for details.

`height`

The height of the frame in CSS pixels. Default is `150`.

`loading`

Indicates how the browser should load the iframe:

- `eager`: Load the iframe immediately, regardless if it is outside the visible viewport (this is the default value).

- `lazy`: Defer loading of the iframe until it reaches a calculated distance from the viewport, as defined by the browser.

`name`

A targetable name for the embedded browsing context. This can be used in the `target` attribute of the `<a>`, `<form>`, or `<base>` elements; the `formtarget` attribute of the `<input>` or `<button>` elements; or the `windowName` parameter in the `window.open()` method.

`referrerpolicy`

Indicates which referrer to send when fetching the frame's resource:

- `no-referrer`: The `Referer` header will not be sent.
- `no-referrer-when-downgrade`: The `Referer` header will not be sent to origins without TLS (HTTPS).
- `origin`: The sent referrer will be limited to the origin of the referring page: its scheme, host, and port.
- `origin-when-cross-origin`: The referrer sent to other origins will be limited to the scheme, the host, and the port. Navigations on the same origin will still include the path.
- `same-origin`: A referrer will be sent for same origin, but cross-origin requests will contain no referrer information.
- `strict-origin`: Only send the origin of the document as the referrer when the protocol security level stays the same (HTTPS→HTTPS), but don't send it to a less secure destination (HTTPS→HTTP).
- `strict-origin-when-cross-origin` (default): Send a full URL when performing a same-origin request, only send the origin when the protocol

security level stays the same (HTTPS→HTTPS), and send no header to a less secure destination (HTTPS→HTTP).

- `unsafe-url`: The referrer will include the origin *and* the path (but not the fragment, password, or username). **This value is unsafe**, because it leaks origins and paths from TLS-protected resources to insecure origins.

`sandbox`

Applies extra restrictions to the content in the frame. The value of the attribute can either be empty to apply all restrictions, or space-separated tokens to lift particular restrictions:

- `allow-downloads-without-user-activation` 🧪: Allows for downloads to occur without a gesture from the user.
- `allow-downloads`: Allows for downloads to occur with a gesture from the user.
- `allow-forms`: Allows the resource to submit forms. If this keyword is not used, form submission is blocked.
- `allow-modals`: Lets the resource open modal windows .
- `allow-orientation-lock`: Lets the resource lock the screen orientation.
- `allow-pointer-lock`: Lets the resource use the Pointer Lock API.
- `allow-popups`: Allows popups (such as `window.open()`, `target="_blank"`, or `showModalDialog()`). If this keyword is not used, the popup will silently fail to open.
- `allow-popups-to-escape-sandbox`: Lets the sandboxed document open new windows without those windows inheriting the sandboxing. For example,

this can safely sandbox an advertisement without forcing the same restrictions upon the page the ad links to.

- `allow-presentation`: Lets the resource start a presentation session.
- `allow-same-origin`: If this token is not used, the resource is treated as being from a special origin that always fails the same-origin policy (potentially preventing access to data storage/cookies and some JavaScript APIs).
- `allow-scripts`: Lets the resource run scripts (but not create popup windows).
- `allow-storage-access-by-user-activation` : Lets the resource request access to the parent's storage capabilities with the Storage Access API.
- `allow-top-navigation`: Lets the resource navigate the top-level browsing context (the one named `_top`).
- `allow-top-navigation-by-user-activation`: Lets the resource navigate the top-level browsing context, but only if initiated by a user gesture.

> **Note:**
>
> - When the embedded document has the same origin as the embedding page, it is **strongly discouraged** to use both `allow-scripts` and `allow-same-origin`, as that lets the embedded document remove the `sandbox` attribute — making it no more secure than not using the `sandbox` attribute at all.
> - Sandboxing is useless if the attacker can display content outside a sandboxed `iframe` — such as if the viewer opens the frame in a new tab. Such content should be also served from a *separate*

> *origin* to limit potential damage.
>
> - The `sandbox` attribute is unsupported in Internet Explorer 9 and earlier.

`src`

The URL of the page to embed. Use a value of `about:blank` to embed an empty page that conforms to the same-origin policy. Also note that programmatically removing an `<iframe>`'s src attribute (e.g. via `Element.removeAttribute()`) causes `about:blank` to be loaded in the frame in Firefox (from version 65), Chromium-based browsers, and Safari/iOS.

`srcdoc`

Inline HTML to embed, overriding the `src` attribute. If a browser does not support the `srcdoc` attribute, it will fall back to the URL in the `src` attribute.

`width`

The width of the frame in CSS pixels. Default is `300`.

Deprecated attributes

These attributes are deprecated and may no longer be supported by all user agents. You should not use them in new content, and try to remove them from existing content.

`align`

The alignment of this element with respect to the surrounding context.

`frameborder`

The value `1` (the default) draws a border around this frame. The value `0` removes the border around this frame, but you should instead use the CSS property `border` to control `<iframe>` borders.

`longdesc`
A URL of a long description of the frame's content. Due to widespread misuse, this is not helpful for non-visual browsers.

`marginheight`
The amount of space in pixels between the frame's content and its top and bottom borders.

`marginwidth`
The amount of space in pixels between the frame's content and its left and right borders.

`scrolling`
Indicates when the browser should provide a scrollbar for the frame:

- `auto`: Only when the frame's content is larger than its dimensions.
- `yes`: Always show a scrollbar.
- `no`: Never show a scrollbar.

Scripting

Inline frames, like `<frame>` elements, are included in the `window.frames` pseudo-array.

With the DOM `HTMLIFrameElement` object, scripts can access the `window` object of the framed resource via the `contentWindow` property. The `contentDocument` property refers to the `document` inside the `<iframe>`, same as `contentWindow.document`.

From the inside of a frame, a script can get a reference to its parent window with `window.parent`.

Script access to a frame's content is subject to the same-origin policy. Scripts cannot access most properties in other `window` objects if the script was loaded from a different origin, including scripts inside a frame accessing the frame's parent. Cross-origin communication can be achieved using `Window.postMessage()`.

Positioning and scaling

As a replaced element, the position, alignment, and scaling of the embedded document within the `<iframe>` element's box, can be adjusted with the `object-position` and `object-fit` properties.

Examples

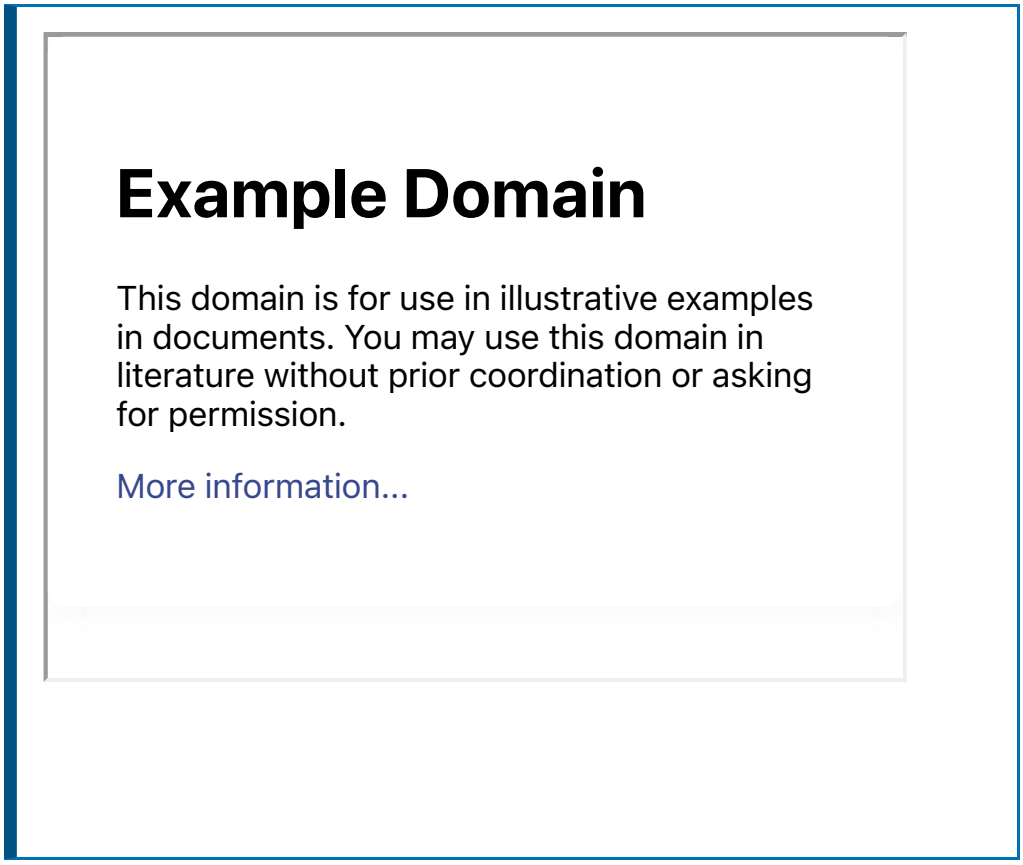
A simple <iframe>

This example embeds the page at https://example.org in an iframe.

HTML

```
<iframe src="https://example.org"
            title="iframe Example 1" width="400"
</iframe>
```

Result



Accessibility concerns

People navigating with assistive technology such as a screen reader can use the `title` attribute on an `<iframe>` to label its content. The title's value should concisely describe the embedded content:

```
<iframe title="Wikipedia page for Avocados" src="
```

Without this title, they have to navigate into the `<iframe>` to determine what its embedded content is. This context shift can be confusing and time-consuming, especially for pages with multiple `<iframe>`s and/or if embeds contain interactive content like video or audio.

Specifications

Specification

HTML Standard (HTML)

the-iframe-element

Browser compatibility

Report problems with this compatibility data on GitHub

	Chrome	Edge	Firefox	Internet Explorer	Opera	Safari	WebView Android	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet
`iframe`	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
`align`	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
`allow`	60	79	74	**No**	47	11	60	60	**No**	44	11	8.0
`allowfullscreen`	27	79	18	11 **-)**	15	7	37	27	18	14	7	1.5
`allowpaymen-trequest`	**No**	**No**	**56**	**No**	**No**	**No**	**No**	**No**	**56**	**No**	**No**	**No**
	79	79	71	**No**	66	14	79	79	79	57	14	12

Aspect ratio computed from `width` and `height` attributes			—									
External protocol URLs blocked	?	?	67	?	?	?	?	?	67	?	?	?
`frameborder`	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
`height`	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
`loading`	77	79	**No** —	**No**	64	**No** —	77	77	**No** —	55	**No** —	12
`longdesc`	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
`marginheight`	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
`marginwidth`	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
`name`	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
`referrerpolicy`	51	79	50	**No**	38	11	51	51	50	41	**No**	7.1
`sandbox`	4	12	17	10	15	5	Ye	Ye	17	?	4.1	Ye
`sandbox="allow-downloads"`	83	83	82	**No**	?	**No**	83	83	82	?	**No**	13

sandbox="allow-modals"	?	?	49	**No**	?	?	?	?	49	?	?	?
sandbox="allow-popups"	Ye	18	28	?	Ye	?	Ye	Ye	27	?	?	Ye
sandbox="allow-popups-to-escape-sandbox"	46	79	49	**No**	32	?	46	46	49	32	?	5.0
sandbox="allow-presentation"	53	79	50	**No**	40	?	**No**	53	50	41	?	6.0
sandbox="allow-same-origin"	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
sandbox="allow-storage-access-by-user-activation"	**No**	**No**	**65**	**No**	**No**	11	**No**	**No**	**65**	**No**	11	**No**
sandbox="allow-top-navigation-by-user-activation"	58	79	79	**No**	45	11	58	58	79	43	?	7.0
scrolling	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
src	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye
srcdoc	20	79	25	**No**	15	6	37	25	25	?	?	1.5

`width`	1	12	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye	Ye

Full support

No support

Compatibility unknown

Experimental. Expect behavior to change in the future.

Non-standard. Check cross-browser support before using.

Deprecated. Not for use in new websites.

See implementation notes.

User must explicitly enable this feature.

-x-

Requires a vendor prefix or different name for use.

See also

- Privacy, permissions, and information security

Last modified: Oct 3, 2021, by MDN contributors