

# Document: DOMContentLoaded event



Baseline Widely available



The **DOMContentLoaded** event fires when the HTML document has been completely parsed, and all deferred scripts ( `<script defer src="...">` and `<script type="module">` ) have downloaded and executed. It doesn't wait for other things

 mdn web docs

**DOMContentLoaded** does not wait for stylesheets to load, however deferred scripts *do* wait for stylesheets, and the **DOMContentLoaded** event is queued after deferred scripts. Also, scripts which aren't deferred or async (e.g., `<script>` ) will wait for already-parsed stylesheets to load.

A different event, **load**, should be used only to detect a fully-loaded page. It is a common mistake to use **load** where **DOMContentLoaded** would be more appropriate.

This event is not cancelable.

## Syntax

Use the event name in methods like `addEventListener()`.

JS



```
addEventListener("DOMContentLoaded", (event) => { })
```



**Note:** There is no `onDOMContentLoaded` event handler property for this event.

# Event type

A generic [Event](#).

## Examples

### Basic usage

JS



```
document.addEventListener("DOMContentLoaded", (event) => {  
  console.log("DOM fully loaded and parsed");  
});
```

### Delaying DOMContentLoaded

HTML



```
<script>  
  document.addEventListener("DOMContentLoaded", (event) => {  
    console.log("DOM fully loaded and parsed");  
  });  
  
  for (let i = 0; i < 1_000_000_000; i++);  
  // This synchronous script is going to delay parsing of the DOM,  
  // so the DOMContentLoaded event is going to launch later.  
</script>
```

### Checking whether loading is already complete

`DOMContentLoaded` may fire before your script has a chance to run, so it is wise to check before adding a listener.

JS





```
function doSomething() {  
  console.info("DOM loaded");  
}  
  
if (document.readyState === "loading") {  
  // Loading hasn't finished yet  
  document.addEventListener("DOMContentLoaded", doSomething);  
} else {  
  // `DOMContentLoaded` has already fired
```

```
doSomething();  
}
```

**Note:** There's no race condition here — it's not possible for the document to be loaded between the `if` check and the `addEventListener()` call. JavaScript has run-to-completion semantics, which means if the document is loading at one particular tick of the event loop, it can't become loaded until the next cycle, at which time the `doSomething` handler is already attached and will be fired.

## Live example

### HTML

```
HTML   Play  
  
<div class="controls">  
  <button id="reload" type="button">Reload</button>  
</div>  
  
<div class="event-log">  
  <label for="eventLog">Event log:</label>  
  <textarea  
    readonly  
    class="event-log-contents"  
    rows="8"  
    cols="30"  
    id="eventLog"></textarea>  
</div>
```

### JavaScript

```
JS   Play  
  
const log = document.querySelector(".event-log-contents");  
const reload = document.querySelector("#reload");  
  
reload.addEventListener("click", () => {  
  log.textContent = "";  
  setTimeout(() => {  
    window.location.reload(true);  
  }, 200);  
});
```

```
});

window.addEventListener("load", (event) => {
  log.textContent += "load\n";
});

document.addEventListener("readystatechange", (event) => {
  log.textContent += `readystatechange: ${document.readyState}\n`;
});

document.addEventListener("DOMContentLoaded", (event) => {
  log.textContent += "DOMContentLoaded\n";
});
```

Result

⌂ Play

Reload

Event log:  
readystatechange: interactive  
DOMContentLoaded  
readystatechange: complete  
load

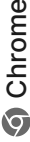
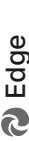
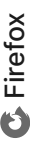

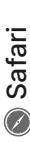


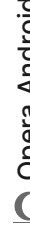




Specifications

Specification
<a href="#">HTML</a> <a href="#"># stop-parsing</a>

Browser compatibility

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

	Desktop					Mobile						
	Chrome	Firefox	Safari	Edge	Opera	Chrome	Firefox	Safari	Edge	Opera	Android	iOS

												
DOMContentLoaded event	✓ 1	✓ 12	✓ 1	✓ 9	✓ 3.1	✓ 18	✓ 4	✓ 10.1	✓ 2	✓ 1	✓ 4.4	✓ 2

Tip: you can click/tap on a cell for more information.

✓ | Full support

## See also

- Related events: [load](#), [readystatechange](#), [beforeunload](#), [unload](#)

## Help improve MDN

Was this page helpful to you?

 Yes

 No

[Learn how to contribute.](#)



This page was last modified on May 2, 2025 by [MDN contributors](#).