

Document: readyState property

The `Document.readyState` property describes the loading state of the `document`. When the value of this property changes, a `readystatechange` event fires on the `document` object.

Value

The `readyState` of a document can be one of following:

loading

The `document` is still loading.

interactive

The document has finished loading and the document has been parsed but sub-resources such as scripts, images, stylesheets and frames are still loading. The state indicates that the `DOMContentLoaded` event is about to fire.

complete

The document and all sub-resources have finished loading. The state indicates that the `load` event is about to fire.

Examples

Different states of readiness

```
JS
switch (document.readyState) {
  case "loading":
    // The document is loading.
    break;
  case "interactive": {
    // The document has finished loading and we can access DOM elements.
    // Sub-resources such as scripts, images, stylesheets and frames are still loading.
    const span = document.createElement("span");
    span.textContent = "A <span> element.";
    document.body.appendChild(span);
    break;
  }
  case "complete":
```

```
// The page is fully loaded.
console.log(
  `The first CSS rule is: ${document.styleSheets[0].cssRules[0].cssText}`,
);
break;
}
```

readystatechange as an alternative to DOMContentLoaded event

```
JS
// Alternative to DOMContentLoaded event
document.onreadystatechange = () => {
  if (document.readyState === "interactive") {
    initApplication();
  }
};
```

readystatechange as an alternative to load event

```
JS
// Alternative to load event
document.onreadystatechange = () => {
  if (document.readyState === "complete") {
    initApplication();
  }
};
```

readystatechange as event listener to insert or modify the DOM before DOMContentLoaded

```
JS
document.addEventListener("readystatechange", (event) => {
  if (event.target.readyState === "interactive") {
    initLoader();
  } else if (event.target.readyState === "complete") {
    initApp();
  }
});
```

Specifications

Specification

[HTML Standard](#)

[# current-document-readiness](#)

Browser compatibility

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

	<div></div>						
	Chrome	Edge	Firefox	Opera	Safari	Chrome Android	Firefox for Android
readyState	✓ Chrome 1	✓ Edge 12	✓ Firefox 3.6	✓ Opera 11 *	✓ Safari 1	✓ Chrome 18 Android	✓ Firefox 4 for Android

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

✓ Full support * See implementation notes.

See also

- Related events:
 - `readystatechange`
 - `DOMContentLoaded`
 - `load`

This page was last modified on Jul 7, 2023 by [MDN contributors](#).