Document: lastModified property

- Value
- Examples
  - Simple usage
  - Transforming lastModified into a Date object
  - Transforming lastModified into milliseconds
- Notes
- Specifications
- Browser compatibility

The `lastModified` property of the Document interface returns a string containing the date and local time on which the current document was last modified.

Value

A string.

Examples

Simple usage

This example alerts the value of `lastModified`.

```
alert(document.lastModified);
// returns: Tuesday, December 16, 2017 11:09:42
```

Transforming lastModified into a Date object

This example transforms `lastModified` into a Date object.

```
let oLastModif = new Date(document.lastModified);
```

Transforming lastModified into milliseconds

This example transforms `lastModified` into the number of milliseconds since January 1, 1970, 00:00:00, local time.

```
let nLastModif = Date.parse(document.lastModified);
```

Notes

Note that as a string, `lastModified` cannot *easily* be used for comparing the modification dates of documents. Here is a possible example of how to show an alert message when the page changes (see also: JavaScript cookies API):

```
// Match 'timestamp' in 'last_modif=timestamp'
// e.g. '1687964614822' in 'last_modif=1687964614822'
const pattern = /last_modif\s*=\s*([^;]*)/;

if (
  Date.parse(document.lastModified) >
  (parseFloat(document.cookie.match(pattern)?.[1]) || 0)
) {
  document.cookie = `last_modif=${Date.now()}; expires=Fri, 31 Dec 9999 23:59:59 GMT; path=${
    location.pathname
  }`;
  alert("This page has changed!");
}
```

...the same example, but skipping the first visit:

```
const pattern = /last_modif\s*=\s*([^;]*)/;

const lastVisit = parseFloat(document.cookie.replace(pattern, "$1"));
const lastModif = Date.parse(document.lastModified);

if (Number.isNaN(lastVisit) || lastModif > lastVisit) {
  document.cookie = `last_modif=${Date.now()}; expires=Fri, 31 Dec 9999 23:59:59 GMT; path=${
    location.pathname
  }`;

  if (isFinite(lastVisit)) {
    alert("This page has been changed!");
  }
}
```

If you want to know whether an *external* page has changed, you can make a HEAD request using the fetch() API, and examine the Last-Modified response header.

Specifications

Browser compatibility

□□□ #3
□ □□□ □□□ 2026-10-31 03:27:53 UTC
□ cocobook-sync □□□ 2026-11-04 17:33:51 UTC