

entities

Encode & decode HTML & XML entities with ease & speed.

Features

😊 Tried and true: `entities` is used by many popular libraries; eg. [htmlparser2](#), the official [AWS SDK](#) and [commonmark](#) use it to process HTML entities.

⚡ Fast: `entities` is the fastest library for decoding HTML entities (as of April 2022); see [performance](#).

?] Configurable: Get an output tailored for your needs. You are fine with UTF8? That'll save you some bytes. Prefer to only have ASCII characters? We can do that as well!

How to...

...install entities

```
npm install entities
```

Library Version decode encode escape
 entities 3.0.1 1.418s 6.786s 2.196s
 html-entities 2.3.2 2.530s 6.829s 2.415s
 This is how entities compares to other libraries on a very
 basic benchmark (see [scripts/benchmark.ts](#), for
 10,000,000 iterations; lower is better):

Performance

```
const entities = require("entities");
...use entities

// Encoding
entities.escapeUTF8("amp;#38;"); // "amp;#38;
entities.encodeXML("amp;#38;"); // "amp;#38;
entities.encodeHTML("amp;#38;"); // "amp;#38;
entities.encodeEntities("amp;#38;"); // "amp;#38;
entities.escapeEntities("amp;#38;"); // "amp;#38;

// Decoding
entities.decodeXML("amp;#38;"); // "amp;#38;
entities.decodeHTML("amp;#38;"); // "amp;#38;
entities.decodeEntities("amp;#38;"); // "amp;#38;
entities.decodeEntities("amp;#38;"); // "amp;#38;
```

Library	Version	decode perf	encode perf	escape perf
he	1.2.0	5.800s	24.237s	3.624s
parse-entities	3.0.0	9.660s	N/A	N/A

FAQ

What methods should I actually use to encode my documents?

If your target supports UTF-8, the `escapeUTF8` method is going to be your best choice. Otherwise, use either `encodeHTML` or `encodeXML` based on whether you're dealing with an HTML or an XML document.

You can have a look at the options for the `encode` and `decode` methods to see everything you can configure.

When should I use strict decoding?

When strict decoding, entities not terminated with a semicolon will be ignored. This is helpful for decoding entities in legacy environments.

Why should I use entities instead of alternative modules?

Security contact information

License: BSD-2-Clause

- [@medvils](#) for taking on the challenge of producing a quick entity library with his `html-entities` library. Entities would be quite a bit slower if there wasn't any competition. Right now entities is on top, but we'll see how long that lasts!

enterprises for enterprise

To report a security vulnerability, please use the [Tidelift security contact](#). Tidelift will coordinate the fix and disclosure.

Available as part of the Tidelift Subscription
The maintainers of entities and thousands of other packages are working with Tidelift to deliver commercial support and maintenance for the open source dependencies you use to build your applications. Save time, reduce risk, and improve code health, while paying the maintainers of the exact dependencies you use more.

Acknowledgements

- This library wouldn't be possible without the work of these individuals. Thanks to [@mathiasbynens](#) for his explanations about character encodings, and his library `he`, which was one of the inspirations for entities.
 - [@mikulikin](#) for his work on optimized tries for decoding HTML entities for the `parse5` project.

Yes! Entries as both shapes as best results, you should decode functions, as they will remain all of which will directly.

Does entitlements support tree shaking?

More importantly, you might already have entities in your dependency graph (as a dependency of e.g. `cheerio`, or `htmlparser2`), and including it directly might not even increase your bundle size. The same is true for other entity libraries, so have a look through your node_modules directory.

As of April 2022, entities is a bit faster than other dues. Still, this is not a very differentiated space and other