

What is the "module" package.json field for?

Asked 7 years, 9 months ago Modified 2 years, 8 months ago

Viewed 132k times



I've seen some npm packages ([vue for example](#)) have a `module` field in their package.json.

215



But `module` [is not included in the package.json documentation](#) - is this a convention of some kind? Is there documentation for this somewhere?



javascript

npm

package.json

Share

Improve this question

Follow

edited Nov 4, 2021 at 10:16



[mikemaccana](#)

122k ● 110 ● 424 ● 528

asked Mar 10, 2017 at 1:20



[ryanve](#)

52.4k ● 32 ● 105 ● 142

2 convention for ECMAScript modules - here's a good read on the different approaches to identifying ESM's:

[hackernoon.com/...](#) – [hackerrdave](#) Mar 10, 2017 at 1:38



Update: April 2022

213



The `module` field is not officially defined by Node.js and support is not planned. Instead, the Node.js community settled on [package exports](#) which they believe is more versatile.



For practical reasons JavaScript bundlers will continue to support the `module` field. The esbuild docs [explain](#) when to use `module` as well as related fields `main` and `browser`.

Original Answer

Is `"module"` an official npm property or is this a convention of some kind?

It's a [proposal](#), but likely to be supported due to de facto usage.

Is there documentation for this somewhere?

There is, in fact, and it can be found [right here](#) and later removed [here](#).

What's it for?

ES6 Module Interoperability in Node. Additional discussion can be found [here](#) and [here](#). And here's [a blog post](#) from Rich Harris talking more about it.

Share Improve this answer

edited Apr 18, 2022 at 12:05

Follow



bernie

10.4k ● 7 ● 66 ● 99

answered Mar 15, 2017 at 17:59



vhs

10k ● 3 ● 70 ● 77

- 1 Does WebPack have documentation on this somewhere, or just Rollup? – [tech4him](#) Dec 30, 2017 at 1:17
- 2 Looks like they have added it here:
webpack.js.org/guides/author-libraries/#final-steps
– [tech4him](#) Aug 24, 2018 at 23:29
- 3 This never got implemented. Instead they have the `type` field that can be set to `"module"`. – [oligofren](#) May 23, 2021 at 14:53



79



This is used by bundler tools for ESM (ECMAScript Module) detection. The [Rollup documentation](#) says it pretty well:

If your `package.json` file also has a `module` field, ES6-aware tools like [Rollup](#) and [webpack 2](#) will [import the ES6 module version](#) directly.

[This article on Rollup 1.0](#) says it another way:

The `main` field makes sure that Node users using `require` will be served the UMD version. The `module` field is not an official npm feature but a common convention among bundlers to designate how to import an ESM version of our library.

Further discussion of `pkg.module` is on the [Rollup Github Wiki](#) and the [webpack Docs](#).

Share Improve this answer

edited Jan 28, 2019 at 21:45

Follow

answered Nov 28, 2017 at 17:07



Jack Steam

5,229 ● 2 ● 28 ● 40

-
- 4 Should `main` be a minified version of UMD version of the library? Or should it be an unminified one? What about `module` ? I guess that module should not be minified because the user will use its bundler (Webpack, Parcel, etc...) and by using the source file with ES6 module exports the bundler can leverage tree-shaking, correct? – [tonix](#) Nov 20, 2019 at 22:11
-