Parsing JSON with Node.js

How to parse JSON from a string, and how to read a JSON file in Node.js

Published Aug 07 2018

If you have JSON (https://flaviocopes.com/json/) data as part of a string, the best way to parse it is by using the `JSON.parse` method that's part of the JavaScript standard since ECMAScript 5, and it's provided by V8 (https://flaviocopes.com/v8/) , the JavaScript engine that powers Node.js (https://flaviocopes.com/nodejs/) .

Example:

```
const data = '{ "name": "Flavio", "age": 35 }'
try {
  const user = JSON.parse(data)
} catch(err) {
  console.error(err)
}
```

Note that `JSON.parse` is synchronous, so the more the JSON file is big, the more time your program execution will be blocked until the JSON is finished parsing.

You can process the JSON asynchronously by wrapping it in a promise (https://flaviocopes.com/javascript-promises/) and a setTimeout call, which makes sure parsing takes place in the next iteration of the event loop:

```
const parseJsonAsync = (jsonString) => {
  return new Promise(resolve => {
    setTimeout(() => {
      resolve(JSON.parse(jsonString))
    })
  })
}

const data = '{ "name": "Flavio", "age": 35 }'
parseJsonAsync(data).then(jsonData => console.log(jsonData))
```

If your JSON is in a file instead, you first have to read it.

A very simple way to do so is to use `require()`:

```
const data = require('./file.json')
```

Since you used the `.json` extension, `require()` is smart enough to understand that, and parse the JSON in the `data` object.

One caveat is that file reading is synchronous. Plus, the result of the require() call is cached, so if you call it again because you updated the file, you won't get the new contents until the program

exits.

This feature was provided to use a JSON file for the app configuration, and it's a perfectly valid use case.

You can also read the file manually, using `fs.readFileSync` :

```
const fs = require('fs')
const fileContents = fs.readFileSync('./file.json',
'utf8')

try {
   const data = JSON.parse(fileContents)
} catch(err) {
   console.error(err)
}
```

This reads the file synchronously.

You can also read the file asynchronously using `fs.readFile` , and this is the best option. In this case, the file content is provided as a callback, and inside the callback you can process the JSON:

```
const fs = require('fs')

fs.readFile('/path/to/file.json', 'utf8', (err, fileC
ontents) => {
   if (err) {
     console.error(err)
     return
   }
   try {
```

```
    const data = JSON.parse(fileContents)
  } catch(err) {
    console.error(err)
  }
})
```

FREE EBOOKS 👈

**JavaScript
(https://flaviocopes.com/page/javascript-
handbook)**

Python (https://flaviocopes.com/page/python-
handbook/)

**React (https://flaviocopes.com/page/react-
handbook/)**

HTML (https://flaviocopes.com/page/html-
handbook/)

CSS (https://flaviocopes.com/page/css-
handbook/)

Node.js (https://flaviocopes.com/page/node-
handbook/)

Linux (https://flaviocopes.com/page/linux-
commands-handbook/)

C (https://flaviocopes.com/page/c-handbook/)

Express.js (https://flaviocopes.com/page/express-
handbook/)

Next.js (https://flaviocopes.com/page/nextjs-
handbook/)

Vue.js (https://flaviocopes.com/page/vue-
handbook/)

Svelte (https://flaviocopes.com/page/svelte-
handbook/)

Deno (https://flaviocopes.com/page/deno-
handbook/)

**ES5 to ESNext (https://flaviocopes.com/page/es5-
to-esnext/)**

**How to Start a Blog
(https://flaviocopes.com/page/book-blog/)**

APRIL 2022 👀

The Web Development Bootcamp
(https://bootcamp.dev)

You might also like

Node.js Streams (https://flaviocopes.com/nodejs-streams/)

Build an HTTP Server (https://flaviocopes.com/node-http-server/)

How to check if a file exists in Node.js (https://flaviocopes.com/how-to-check-if-file-exists-node/)

How to get the names of all the files in a folder in Node (https://flaviocopes.com/how-to-get-files-names/)

Find the installed version of an npm package (https://flaviocopes.com/npm-know-version-installed/)

How to update Node.js (https://flaviocopes.com/how-to-update-node/)

How to use promises and await with Node.js callback-based functions (https://flaviocopes.com/node-promisify/)

Where does npm install the packages? (https://flaviocopes.com/where-npm-install-packages/)

Writing files with Node (https://flaviocopes.com/node-writing-files/)

How to remove a file with Node.js (https://flaviocopes.com/how-to-remove-file-node/)