Reading files with Node.js

The simplest way to read a file in Node.js is to use the `fs.readFile()` method, passing it the file path and a callback function that will be called with the file data (and the error):

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

fs.readFile('/Users/joe/test.txt', (err, data) => {
  if (err) {
    console.error(err)
    return
  }
  console.log(data)
})
```

Alternatively, you can use the synchronous version `fs.readFileSync()`:

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

try {
  const data = fs.readFileSync('/Users/joe/test.txt', 'utf8')
  console.log(data)
} catch (err) {
  console.error(err)
}
```

The default encoding is utf8, but you can specify a custom encoding using a second parameter.

Both `fs.readFile()` and `fs.readFileSync()` read the full content of the file in memory before returning the data.

This means that big files are going to have a major impact on your memory consumption and speed of execution of the program.

In this case, a better option is to read the file content using streams.

CONTRIBUTORS

EDIT THIS PAGE ON GITHUB

← PREV NEXT →