
NodeJS: Módulos nativos

Gabriel Rodríguez Flores

October 7, 2021

- Import own files
- Import native object
 - FS
 - Path
 - ReadLine
 - URL
- Scripting

Contents

| | | |
|----------|---|----------|
| 1 | Teoría | 3 |
| 1.1 | Separar el código en distintos ficheros | 3 |
| 1.1.1 | Exports | 3 |
| 1.1.2 | Imports | 3 |
| 1.2 | Objetos nativos | 3 |
| 1.2.1 | File System | 3 |
| 1.2.2 | Path | 4 |
| 1.2.3 | Read Line | 4 |
| 1.2.4 | Process | 5 |
| 1.2.5 | URL | 5 |
| 1.2.6 | OS | 5 |
| 1.2.7 | Events | 5 |
| 1.2.8 | Buffer | 5 |
| 1.2.9 | Stream | 5 |
| 1.2.10 | Errors | 5 |
| 1.2.11 | HTTP | 5 |
| 2 | Ejemplos | 6 |
| 2.1 | Ejemplo de importar ficheros propios | 6 |
| 2.2 | Ejemplo FileSystem | 6 |
| 3 | Ejercicios | 6 |
| 4 | Entregables | 7 |
| 4.1 | En clase | 7 |
| 4.2 | Tarea | 7 |

1 Teoría

- Cheatsheet I
- Cheatsheet II

1.1 Separar el código en distintos ficheros

1.1.1 Exports

```
1 module.exports = variableAExportar;  
2 // o  
3 module.exports = {  
4   variable,  
5   funcion,  
6   // ...  
7 };
```

1.1.2 Imports

```
1 const variableDondeImportar = require('./file'); // sin la extensión .  
   js !!  
2 // o  
3 const { variable, funcion } = require('./file');
```

1.2 Objetos nativos

1.2.1 File System

Resumen

- Lectura

```
1 const fs = require('fs')  
2  
3 try {  
4   const data = fs.readFileSync('/Users/joe/test.txt', 'utf8')  
5   console.log(data)  
6 } catch (err) {  
7   console.error(err)  
8 }
```

- Escritura

```
1 const fs = require('fs')
2
3 const content = 'Some content!'
4
5 try {
6   const data = fs.writeFileSync('/Users/joe/test.txt', content)
7   //file written successfully
8 } catch (err) {
9   console.error(err)
10 }
```

- Carpetas

1.2.2 Path

Ejemplos

```
1 const path = require('path')
2
3 const notes = '/users/joe/notes.txt'
4
5 path.dirname(notes) // /users/joe
6 path.basename(notes) // notes.txt
7 path.extname(notes) // .txt
8 path.basename(notes, path.extname(notes)) //notes
9
10 const name = 'joe'
11 path.join('/', 'users', name, 'notes.txt') //'users/joe/notes.txt'
12 path.resolve('joe.txt') //'Users/joe/joe.txt' if run from my home
   folder
13 path.resolve('tmp', 'joe.txt') //'Users/joe/tmp/joe.txt' if run from
   my home folder
14 path.resolve('/etc', 'joe.txt') //'etc/joe.txt'
15 path.normalize('/users/joe/../../test.txt') //'users/test.txt'
```

1.2.3 Read Line

```
1 const readline = require('readline');
2
3 const rl = readline.createInterface({
4   input: process.stdin,
5   output: process.stdout
6 });
7
8 rl.question('What do you think of Node.js? ', (answer) => {
9   // TODO: Log the answer in a database
```

```
10 console.log(`Thank you for your valuable feedback: ${answer}`);  
11  
12 rl.close();  
13 });
```

1.2.4 Process

- `process.env` Variables de entorno
- `process.argv` Argumentos de la invocación del script

1.2.5 URL

1.2.6 OS

Ejemplos

1.2.7 Events

Ejemplos

1.2.8 Buffer

Ejemplos

1.2.9 Stream

Ejemplos

1.2.10 Errors

Ejemplos

1.2.11 HTTP

Próximamente ...

2 Ejemplos

2.1 Ejemplo de importar ficheros propios

- `notes.js`

```
1 // function getNotes() {  
2   const getNotes = function () {  
3     return 'Your notes...';  
4   };  
5  
6   module.exports = getNotes;
```

- `index.js`

```
1 const getNotes = require('./notes');  
2  
3 const msg = getNotes();  
4  
5 console.log(msg);
```

2.2 Ejemplo FileSystem

```
1 const fs = require('fs')  
2 // fs.writeFileSync('notes.txt', 'My name is Andrew.')  
3 fs.appendFileSync('notes.txt', ' I live in Philadelphia.')
```

3 Ejercicios

1. Realizar programa que reciba por argumentos el nombre de un fichero e imprima su contenido
2. Crear un editor de notas con Nodejs

- Tendrá un menú que permita:

1. Crear nueva nota
2. Editar nota existente
3. Eliminar nota

- Cada nota será un fichero de texto, con extensión `.note`
- Para editar una nota, habrá que imprimir la lista de notas disponibles y seleccionar una
- Cuando se está escribiendo una nota, ha de permitir continuar su edición hasta introducir dos veces seguidas una linea en blanco

3. Descomponer una URL extrayendo: protocol, ipAddress, subDomain, domainName, folderTree, targetFile, argumentsFile.

- enlace

4 Entregables

4.1 En clase

- Ejercicio 1 y 3

4.2 Tarea

- Ejercicio 2