Développement Backend avec Node.js

Web School Factory

4A - P2023

2021-2022 Hugo Caillard @Cohars

Summary

Annexes

`index.js`

```
import http from 'http'

const server = http.createServer((req, res) => {
  res.write('Hello world')
  res.end()
})

server.listen('5000')
console.log('server listening on http://localhost:5000')
```

```
$ node index.js
```

NPM

NPM est le `package manager` par défaut de Node.js. Des alternatives existent (yarn, pnpm) mais nous nous concentrerons sur NPM.

NPM sert notamment à :

- Gérer les dépendances de notre projet
- Lancer des scripts pour exécuter son code ou le tester par exemple

---

`package.json`

Dans un dossier vide, lancer ;

```
$ npm init
// or to accept all default values
$ npm init -y
```

Exemple de `package.json`, quelques propriétés importante :

```
{
  "name": "my-project",
  "version": "1.0.0",
  "type": "module",
  "scripts": {
    "start": "node index.js",
    "dev": "npx nodemon index.js",
    "test": "jest"
  }
}
```

Installer des dépendances

```
$ npm install dotenv // or $ npm i dotenv
$ npm install --save-dev jest nodemon // or $ npm i -D jest nodemon
```

`packages.json`

```
{
  "dependencies": {
    "dotenv": "^10.0.0",
  },
  "devDependencies": {
    "jest": "^27.2.3",
    "nodemon": "^2.0.13"
  }
}
```

Documentation officielle

Utiliser un package / module

```
import superagent from 'superagent'

const url = 'https://jsonplaceholder.typicode.com/posts/1'

async function test() {
  try {
    const res = await superagent.get(url)
    console.log(res.body)
  } catch (err) {
    console.log(err)
  }
}

test()
```

Tester ce code

Créer un module

./add.js

```
/**
 * @param {number} a
 * @param {number} b
 * @returns {number}
 */
export function add(a, b) {
  return a + b
}
```

./index.js

```
import { add } from './add.js'

add(1, 2) // 3
```

JSDoc

```
/**
 * @param {number} a
 * @param {number} b
 * @returns {number}
 */
```

JSDoc permet de documenter son code directement dans les commentaires.

Il existe plusieurs syntaxes, celle qui nous intéresse plus particulièrement est celle de TypeScript. C'est un bon moyen de typer son code tout en écrivant du JS "classique".

TypeScript with JSDoc JSDoc support in VS Code