Advanced JavaScript and Node.js

Tan E-Liang

Mar 9, 2019

Hackerschool

Last week: JavaScript basics

Today

1. “Advanced” JavaScript
2. Make a simple web server in Node

Next Week: Build a single-page web app in React

Set Up

1. Node.js version ≥ 8. A Node feature support matrix can be found at node.green.

```
brew install node # macOS
sudo pacman install nodejs # Arch Linux
```

2. These slides and starter code from this GitHub repo. Click the green “Clone or download” button.
3. Some text editor

Recap—JavaScript Fundamentals

```
1 var neverUseVar = 12345678;
2 let mutable = [1, "two", "3"];
3 const immutable = "stone";
4
5 function myFunction(arg1, arg2) {
6     console.log(arg1, arg2);
7 }
8 setInterval(callbackFunction, 300);
```

Today

1. JavaScript
    1.1 Classes
    1.2 Functions as first class citizens
    1.3 Arrow functions (i.e. `(arg1, arg2) => { statements }`)
    1.4 `module.exports` and `require` (and their ES6 equivalents (but not really) `export` and `import`)
    1.5 `async`/`await` and Promises
2. Node
    2.1 Basics
    2.2 REPL
    2.3 NPM
    2.4 Express.js

End product: Build an exam countdown API using the NUSMods API.

Why does Node.js even exist?

Netscape Navigator
Brought interactivity to web pages

Node.js
JavaScript runtime for servers

Electron
Build desktop apps with a Node+browser hybrid

Node.js—Key Components

1. Read-Eval-Print-Loop (REPL)
2. `node` itself
3. Node Package Manager (NPM)

Node REPL

- Launch using `node` in the terminal.
- Type JavaScript code at the prompt.
- Variables can be set and used throughout a REPL session.
- Exit by pressing Ctrl+D

```
> let name = "Chinese"
> name
// Output: name
> console.log("Is it because I'm", name)
// Output: Is it because I'm Chinese
> name = "Malay"
// Press the up arrow key to access your command
↪  history
> console.log("Is it because I'm", name)
// Output: Is it because I'm Malay
```

Node Scripts

Execute scripts by running `node script.js`. Try running this in your terminal:

```
$ cd hackerschool-advanced-js-node/starter
$ node 1_basics.js
```

Advanced JavaScript Part 1

We'll cover some advanced features by running through the scripts in the starter folder.

1. 1_basics.js
   (i) Nested functions
   (ii) String template literals
   (iii) Arrow functions
2. 2_functional.js
   (i) `map`, `reduce`, `filter`
   (ii) Object/array destructuring
   (iii) Object/array rest/spread operator
3. 3_classes.js
   (i) Classes
4. 4.*.js
   (i) `module.exports`, `require`

Node Package Manager (NPM)

NPM is a central collection of published packages/libaries/frameworks/tools.

Run these lines in your terminal:

```
$ cd hackerschool-advanced-js-node/starter/server
$ ls # The folder should contain only server.js
$ npm init # and spam Enter to fill in the prompts. A
  ↪  package.json file should be created.
$ npm install --save express
$ ls # A node_modules folder should be created
```

Per-project configuration file (package.json) contains dependencies and other information. Per-project dependencies are stored in node_modules folder.

These files are in the starter/server directory.

1. 1_express.js
   (i) Basic Express server
   (ii) File server
2. 2_asyncawait.js
   (i) async/await
   (ii) Promises
3. 3_daystoexamserver.js
   Your turn: make an exam countdown using the NUSMods API

Possible Extensions

1. Add an endpoint that counts down the number of days to all exams.

Thank you!

Feedback please!

http://bit.ly/hs-ajs