

Andrew Marcinkevičius

## Section 2

# *Understanding Node.js Apps*



In this Section, we are going to take a look at...

- CommonJS
- npm
- The Event Loop
- Working with streams
- Working with files

# Web Development with Node.js and MongoDB

---

*Andrew Marcinkevičius*

Video 2.1

*CommonJS*



In this Video, we are going to take a look at...

- What CommonJS is
- Organizing modules
- Import/export module

# What Is CommonJS?

- Group with goal to build JavaScript ecosystem
- Ratified modules standard
- Modules keep global scope clean

```
01-main.js
1 'use strict';
2
3 // const fs = require('fs');
4 // const helloString = require('./02-hello-string');
5 // const helloFunction = require('./03-hello-function');
6 // const helloObject = require('./04-hello-object');
7 // const name = require('./04-hello-object').name;
8
9 // console.log(helloString);
10
11 // helloFunction();
12
13 // helloObject.say('Andrew');
14
15 // console.log(name);
16
```



- code
  - 1
  - 2
    - 2.1
      - 01-main.js
      - 02-hello-string.js
      - 03-hello-function.js
      - 04-hello-object.js
    - 2.2
    - 2.3
    - 2.4
    - 2.5
  - 3
  - 4
  - 5
  - 6
  - 7
  - 8
  - export-environment-variables.sh
- scripts

```
01-main.js
1 'use strict';
2
3 const fs = require('fs');
4 const helloString = require('./02-hello-string');
5 // const helloFunction = require('./03-hello-function');
6 // const helloObject = require('./04-hello-object');
7 // const name = require('./04-hello-object').name;
8
9 // console.log(helloString);
10
11 // helloFunction();
12
13 // helloObject.say('Andrew');
14
15 // console.log(name);
16
```

## FOLDERS

- ▼ code
  - ▶ 1
  - ▼ 2
    - ▼ 2.1
      - 01-main.js
      - 02-hello-string.js
      - 03-hello-function.js
      - 04-hello-object.js
    - ▶ 2.2
    - ▶ 2.3
    - ▶ 2.4
    - ▶ 2.5
  - ▶ 3
  - ▶ 4
  - ▶ 5
  - ▶ 6
  - ▶ 7
  - ▶ 8
  - export-environment-variables.sh
- ▶ scripts

01-main.js

02-hello-string.js

```
1 module.exports = 'Hello string';
```

```
2
```



```

1 'use strict';
2
3 const fs = require('fs');
4 const helloString = require('./02-hello-string');
5 // const helloFunction = require('./03-hello-function');
6 // const helloObject = require('./04-hello-object');
7 // const name = require('./04-hello-object').name;
8
9 console.log(helloString);
10
11 // helloFunction();
12
13 // helloObject.say('Andrew');
14
15 // console.log(name);
16

```

ifdattic at ifdattic-mbp13 in ~/projects/nodejs-mongodb-video-course/code/2/2.1 on master [!]

\$ node 01-main.js

Hello string

ifdattic at ifdattic-mbp13 in ~/projects/nodejs-mongodb-video-course/code/2/2.1 on master [!]

\$

Rewind 15s

2:07 / 3:48



Browse Q&A

Add Bookmark

Continue

01-main.js x 03-hello-function.js x Go to Dashboard

```
1 module.exports = function () {
2   console.log('Hello function');
3 }
4
```

2:08 / 3:48

2 Words, Line 4, Column 1; Saved /projects/nodejs-mongodb-video-course/code/2/2.1/03-hello-function.js (UTF-8) 0 misspelled words UTF-8 Unix Spaces: 4 JavaScript (Babel)

```
1 'use strict';
2
3 const fs = require('fs');
4 const helloString = require('./02-hello-string');
5 const helloFunction = require('./03-hello-function');
6 // const helloObject = require('./04-hello-object');
7 // const name = require('./04-hello-object').name;
8
9 console.log(helloString);
10
11 helloFunction();
12
13 // helloObject.say('Andrew');
14
15 // console.log(name);
16
```

## FOLDERS

- ▼ code
  - ▶ 1
  - ▼ 2
    - ▼ 2.1
      - 01-main.js
      - 02-hello-string.js
      - 03-hello-function.js
      - 04-hello-object.js
    - ▶ 2.2
    - ▶ 2.3
    - ▶ 2.4
    - ▶ 2.5
  - ▶ 3
  - ▶ 4
  - ▶ 5
  - ▶ 6
  - ▶ 7
  - ▶ 8
  - export-environment-variables.sh
- ▶ scripts

01-main.js × 04-hello-object.js ×

1



```
1 'use strict';
2
3 const fs = require('fs');
4 const helloString = require('./02-hello-string');
5 const helloFunction = require('./03-hello-function');
6 const helloObject = require('./04-hello-object');
7 // const name = require('./04-hello-object').name;
8
9 console.log(helloString);
10
11 helloFunction();
12
13 // helloObject.say('Andrew');
14
15 // console.log(name);
16
```



```
1 'use strict';
2
3 const fs = require('fs');
4 const helloString = require('./02-hello-string');
5 const helloFunction = require('./03-hello-function');
6 const helloObject = require('./04-hello-object');
7 // const name = require('./04-hello-object').name;
8
9 console.log(helloString);
10
11 helloFunction();
12
13 helloObject.say('Andrew');
14
15 // console.log(name);
16
```

```
ifdattic at ifdattic-mbp13 in ~/projects/nodejs-mongodb-video-course/code/2/2.1 on master [!]
```

```
$ node 01-main.js
```

```
Hello string
```

```
Hello function
```

```
Hello Andrew
```

```
ifdattic at ifdattic-mbp13 in ~/projects/nodejs-mongodb-video-course/code/2/2.1 on master [!]
```

```
$ █
```

```
1 'use strict';
2
3 const fs = require('fs');
4 const helloString = require('./02-hello-string');
5 const helloFunction = require('./03-hello-function');
6 const helloObject = require('./04-hello-object');
7 const name = require('./04-hello-object').name;
8
9 console.log(helloString);
10
11 helloFunction();
12
13 helloObject.say('Andrew');
14
15 // console.log(name);
16
```

```
1 'use strict';
2
3 const fs = require('fs');
4 const helloString = require('./02-hello-string');
5 const helloFunction = require('./03-hello-function');
6 const helloObject = require('./04-hello-object');
7 const name = require('./04-hello-object').name;
8
9 console.log(helloString);
10
11 helloFunction();
12
13 helloObject.say('Andrew');
14
15 console.log(name);
16
```



```
ifdattic at ifdattic-mbp13 in ~/projects/nodejs-mongodb-video-course/code/2/2.1 on master [!]
```

```
$ node 01-main.js
```

```
Hello string
```

```
Hello function
```

```
Hello Andrew
```

```
World
```

```
ifdattic at ifdattic-mbp13 in ~/projects/nodejs-mongodb-video-course/code/2/2.1 on master [!]
```

```
$ █
```

# Importing Modules

- Relative path: `require('./hello')`
- Absolute path: `require('/home/world/hello')`
- Name: `require('hello')`
  - Looks at core libraries or "node\_modules" directory



Next Video

npm