



JÖNKÖPING UNIVERSITY

*School of Engineering*

---

# NODE.JS

**Peter Larsson-Green**

Jönköping University

Spring 2018

# WHAT IS IT?

A JavaScript *Run-time Environment*.

- Run-time: Something that can execute code.
- Environment: Adds things your code can use in addition to JavaScript, such as:
  - Access to the file system.
  - Access to the network.

# INSTALLING NODE.JS

Download and install from: <https://nodejs.org/en>

# JAVASCRIPT VERSIONS

JavaScript: 1995 (used in Netscape)

JScript: 1996 (used in IE3)

ECMAScript 1: 1997

ECMAScript 2: 1998 (specification re-written)

ECMAScript 3: 1999

ECMAScript 4: Abandoned.

ECMAScript 5: 2009

ECMAScript 5.1: 2011 (specification re-written)

ECMAScript 6: 2015 ("ECMAScript 2015")

ECMAScript 7: 2016 ("ECMAScript 2016")

ECMAScript 8: 2017 ("ECMAScript 2017")

ECMAScript 9: 2018 ("ECMAScript 2018") - <https://www.ecma-international.org/publications/standards/Ecma-262.htm>

Which features you can use  
in Node.js: <http://node.green>

# RUNNING NODE.JS

Run Node.js using the `node` program in a shell:

- `node`
  - Enters a "read eval print"-loop.
- `node a-filename.js`
  - Executes the JavaScript code in `a-filename.js`.

# HELLO WORLD

```
console.log("Hello, World!")
```

- `console` is not part of the language, but the environment.

# NODE.JS MODULES

Functionalities in Node.js are separated into different modules, e.g.:

- File System, `fs`, for working with files.
- HTTP, `http`, for working with HTTP.

DOCS: <https://nodejs.org/dist/latest-v10.x/docs/api/>

Needs to explicitly be included before used.

```
const fs = require("fs")
```

- Modules as specified in *ECMAScript* **are not** supported.
- Modules as specified in *CommonJS* **are** supported.



# COMMONJS

Specifies how one JS file can export values other JS files can import.

```
function adder(number) {  
    return number + 1  
}  
module.exports = adder
```

adder.js

```
const adder = require('./adder')
```

```
const six = adder(5)
```

main.js

# COMMONJS

Specifies how one JS file can export values other JS files can import.

```
function addThree (number) {  
  return number + 3  
}  
  
function addFive (number) {  
  return number + 5  
}  
  
exports.addThree = addThree  
exports.addFive = addFive
```

adders.js

```
const adders = require ('./adders')  
  
const four = adders.addThree (1)  
const six = adders.addFive (1)  
  
main.js
```