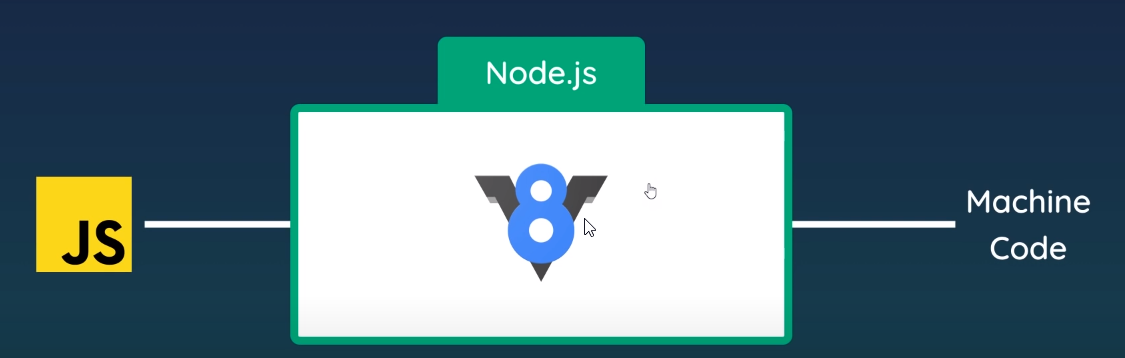
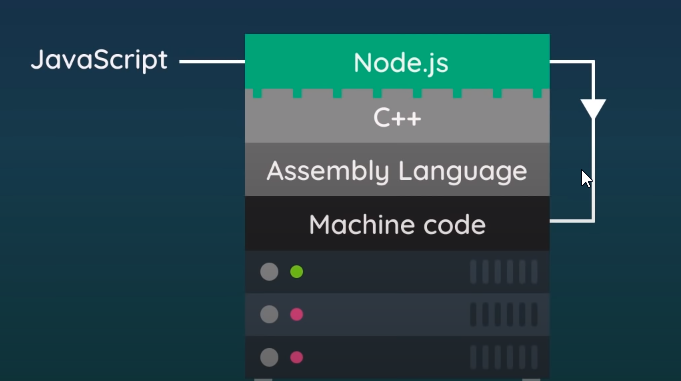
1. Node.js

Node allow to run javascript on the server



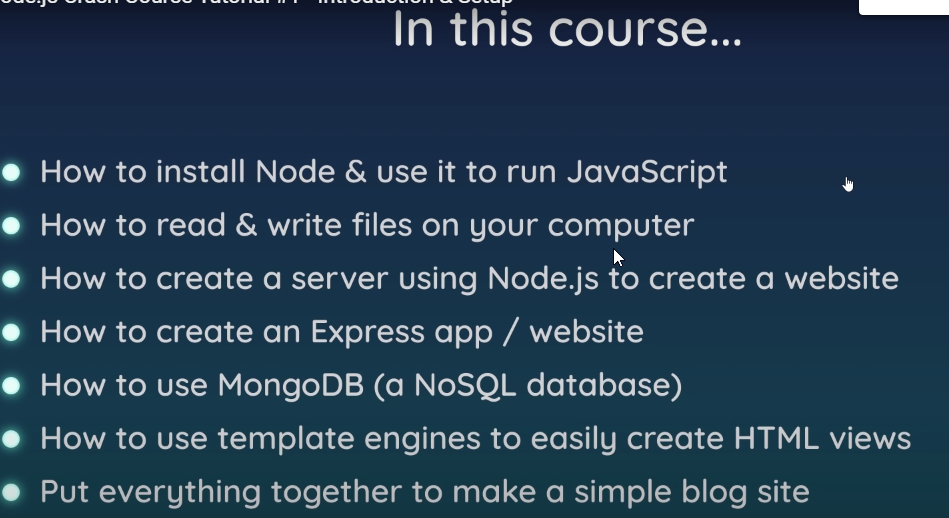
V8 engine compile javascript to understands javascript in the Browser. V8 is made on C++ but outside browser we can use nodejs to compile javascript





The goal of nodejs is to allow the user make some activities in the client explorer and run javascript in the server to request some process be running. (Connect database, request html-css, ect)

1. Course. Pensum



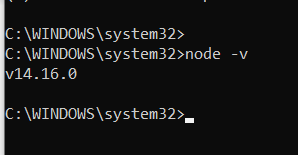
1. Installing Node.js

<https://Nodejs.org/en/>



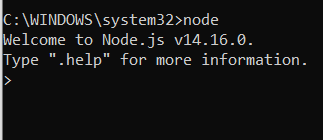
Check version installed (command prompt):

Node -v



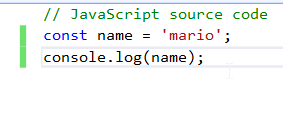
Start process with:

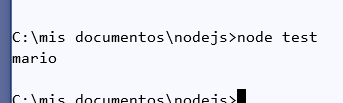
Node



The command node + file.js will execute the program

Node test.js

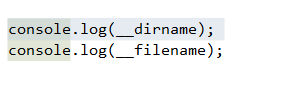
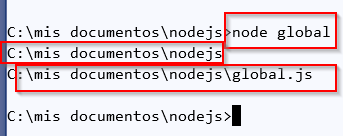




1. Dirname and filename

console.log(\_\_dirname); => absolute path current folder that this file is in

console.log(\_\_filename);=> absolute path of the folder with the file name added on as well



The first console only provides path of folder, the second provides folder + file as well

1. Module and require
2. How to import files into module file?

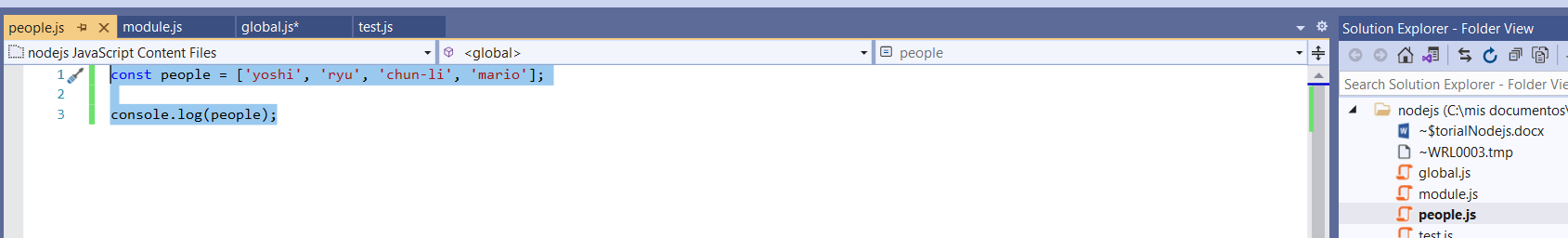
I have 2 files:

* People.js
* Module.js

In people, I have an array

const people = ['yoshi', 'ryu', 'chun-li', 'mario'];

console.log(people);



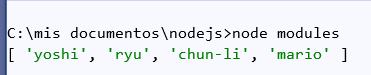
I want to import into module.js. We use require command

In the module file we have

Const xyz = require(‘./people’);

“./” is a relative path and it will look in the same directory where we have module file

If I run node modules



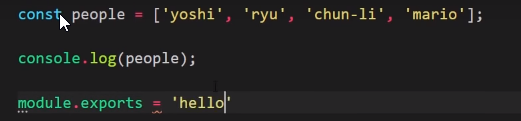
1. Export module

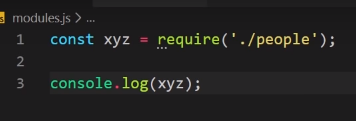
This is made to export variables

In the file people the const people is empty if we try to export to modules but if we use this command into the people.js file:

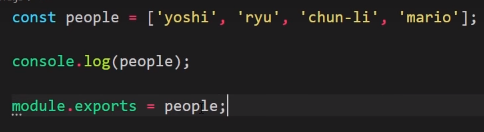
**module.exports** = ‘hello’;

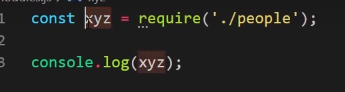
Then, when I use **console.log(xyz);** I will get hello



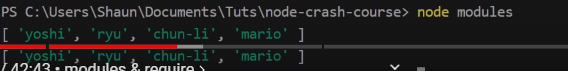


But if I say **module.exports**=people; I would export the array to the file modules.js





Result:



1. We can export several objects

const people = ['yoshi', 'ryu', 'chun-li', 'mario'];

const ages = [20, 25, 30, 35];

console.log(people);

//module.exports = people;

module.exports = {

people, ages

};

And

const xyz = require('./people');

console.log(xyz);

or

const xyz = require('./people');

console.log(xyz.people, xyz.ages);

I also can define(Import):

const {people, ages} = require('./people');

console.log(people, ages); people and ages should be same name of properties in the other file

1. Modules embedded

Const os = require(‘os’); this is embedded in nodejs

console.log(os.platform(), os.homedir());

1. File System
2. Read file

//import system module named as fs

const fs = require('fs');

//reading file

//coming 2 arguments, one is string the second is a function which is triggered when the path is completed

//the path is the relative path where the file is located

//the function is asynchronous

//the function contains 2 values, err and data

fs.readFile('./docs/blog1.txt', (err, data) => {

if (err) {

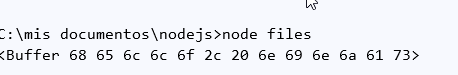
console.log(err);

}

console.log(data);

});

Once we executed, we get a buffer



The buffer coming with this command console.log(data);

If we want to see the string date, we use this option:

Console.log(data.toString());

1. Write file

//import system module named fs

const fs = require('fs');

//three arguments

//first argument relative path

//second is text we want to write

//third call back function

fs.writeFile('./docs/blog1.txt', 'hello world', () => {

console.log('file was written');

});

1. Create folder

//directory

//we create in the current directory

fs.mkdir('./assets', (err) => {

if (err) {

console.log(err);

}

console.log('folder created');

});

We check if the folder exist

if (!fs.existsSync('./assets')) {

fs.mkdir('./assets', (err) => {

if (err) {

console.log(err);

}

console.log('folder created');

});

} else {

fs.rmdir('/assets', (err) => {

if (err) {

console.log(err)

}

console.log('folder deleted');

})

}

For delete files we use fs.unlink function

if (fs.existsSync('./docs/deleteme.txt')) {

fs.unlink('./docs/deleteme.txt', (err) => {

if (err) {

console.log(err)

}

console.log('file deleted');

})

}

1. Streams

We create the path from where we read the buffer with

const readStream = fs.createReadStream('./docs/blog3.txt');

We create a function using readStream.on which is an event which receive the buffer from the path and send chunk of ‘data’

const fs = require('fs');

const readStream = fs.createReadStream('./docs/blog3.txt'); //we set friom where we read data

// on is like an event which listens from 'data'

//read

readStream.on('data', (chunk) => {

console.log('---- NEW CHUNK ----');

console.log(chunk);

});

//Write

readStream.on('data', (chunk) => {

console.log('---- NEW CHUNK ----');

console.log(chunk);

WriteStream.write('\nNEW TEST\n');

WriteStream.write(chunk);

});

We can replace the whole function above with:

readStream.pipe(writeStream);