NodeJS

# Introduction to NodeJS

## Callback function

When one executing a task that takes some time to be executed, it is a good practice to launch that task asynchronously. In this way, other tasks can be executed at the same time and when the “long” task is complete, the callback function will return a value. The snippet code below, show the use of a callback function.

|  |
| --- |
| import fs from ‘fs’;  const phoneNumber = (err, data) => {  console.log(‘data:’, data);  }  fs.readdir(‘/’, phoneNumber);  console.log(‘this become after’); |

## Nodemon module

This module needs to be installed using the terminal. Nodemon allows to run automatically the script when changes are made.

|  |
| --- |
| yarn add nodemon global |

## Reading and Writing files

Access the filesystem using the function readfile. readfile is an asynchronous function and accepts a callback. Callback function can be a defined function or an anonymous function. In order to read the file, it is necessary to decode the file, using the 'utf-8' char code. In this case, before use the read JSON file, it is necessary to parse to a JSON object.

|  |
| --- |
| fs.readFile('./data.json', 'utf-8', (err, data) => {  const info = JSON.parse(data);  console.log(info, info.name);  }); |

Another way to read a JSON file is by simple require the file. This way the read file is already parsed to a JSON object.

|  |
| --- |
| const data = require('./data.json');  console.log(data, data.name); |

To write a file one uses the writeFile function (from fs module). This function accepts three parameters. The first ones the name of the file, the second one is the data to be written to the file and the third one is the callback function. Be careful, the data to be written to the file must be parsed to string before being passed as an argument to the function.

|  |
| --- |
| const fileToWrite = {  data: 'bla',  };  fs.writeFile('newData.json', JSON.stringify(fileToWrite), (err) => {  if (err) throw err;  console.log('The file has been saved');  }); |