**Node.js Basics -** [Dias Utsman](https://medium.com/@utsmand91?source=post_page-----8c620d4f3bf4--------------------------------) Jun 20, 2022

Before I’m going to explain to you about basics of Node.js, we need to prepare 2 tools that are Text Editor and Node.js. For the Text Editor, I recommend you to use VSCode as it’s the most popular and free to use. Besides that, this text editor has a lot of plugins so that its functionalities can be rich. To download it, please visit [the visual studio code download page](https://code.visualstudio.com/download). For Node.js, you can download it [here](https://nodejs.org/en/). Make sure to download version 12 or higher and to check it you can type:

node -v

**What is Node.js?**

Node.js is a javascript runtime that capable of running javascript code outside the browser. Node.js is just like the gate for the javascript developer to develop a system outside the browser. JavaScript is a multi-platform language that many developers use. The popularity of JavaScript has skyrocketed! From 2014 to 2020 JavaScript became the number one programming language that was widely used by developers.

**Node.js Global Object**

In javascript, there is an object that can control browser functionalities such as visiting pages, reloading, closing tabs, and displaying alert dialogs.

It’s the same in node js, it’s added a global object to give more functionalities to javascript. For example, you can use it to see the CPU that is used in the computer, file modularization, print value to the console, etc.

Object window pada browser dan object global di node.js is *Global Object* so that you can use its methods and properties without having to explicitly refer to it.

There are objects in node.js that are called ‘true globals’. That is because these objects are added by the node.js and not from the javascript.

Here is the list:

* global: The global namespace. Any member in this object can be accessed on the global scope.
* process: provides interaction with running Node.js processes.
* console: provides various STDIO functionality.
* setTimeout, clearTimeout, setInterval, clearInterval.

There are also some objects that are ‘pseudo-globals’. This object will not be printed when you print Object.getOwnPropertyName(global). This is because they are not a direct member of a global object but inherited from module scope. Because in Node.js all JavaScript files are modules, so pseudo-globals can be accessed like global objects.

Here is the list:

* module: used for the modularization system in Node.js.
* \_\_filename: keyword to get the location of the executable JavaScript file. This keyword is not available in Node.js REPL.
* \_\_dirname: keyword to get the root directory of the executable JavaScript file.
* require: used to import JavaScript modules.

**Process Object**

Process object contains functions and properties that can provide information about currently running processes.

Here is the list of some functions and properties in the process object.

* process.env: property to store values or get information about the environment used while the process is in progress. Examples: process.env.PWD provides information about the location of the process that is in progress. This property is sometimes used to determine if the server is running when in production or development.
* process.memoryUsage(): property to get information about CPU usage while the process is running.
* Etc.

**Modularization**

Modularization in programming is a technique of separating code into modules that are independent but can be used with each other to form a complex program. This splitting of code into separate modules can make JavaScript code easier to organize.

To export it use: module.exports.

const coffee = {name: ‘Tubruk’,price: 15000,}module.exports = coffee;

To import, it uses require() and then passed in the path to that file relative to this file.

const coffee = require(‘./coffee’);console.log(coffee);/\*\*\* node app.js\*\* output:\* { name: ‘Tubruk’, price: 15000 }\*/

**Node Package Manager**

This is used to download the external packages from [https://www.npmjs.com](https://www.npmjs.com/)/.

That’s all I could say about node.js basics. If you want to learn more, go to <https://www.w3schools.com/nodejs/default.asp>. On that website, you can learn about node.js in a simpler way by seeing examples and implementing them. This has been more than 500 words. Hope this is useful for you.