**#1 What’s node.js and Why we use it –**

[Ruckaiya Mira](https://medium.com/@ruckaiya.awf5?source=post_page-----9c06105bbcc9--------------------------------) Jul 1, 2022

This is a Node.js, Express.js, and MongoDB tutorial series of articles for beginners who want to enhance their skill on Node.js and step toward back-end Web Development.

In this Node.js series, I will cover Node.js, Express.js, and MongoDB from scratch. Most importantly this node.js series is absolutely beginner-friendly. The only prerequisite is you must know JavaScript.

I will publish a new article every two days and little by little you will learn everything there is to know about Node.js.

Follow me on medium: <https://medium.com/@ruckaiya.awf5>

Now before diving into some node.js code let’s do a high level overview of what node.js actually is, what we use it for, and why we use node instead of other technologies.

So the official definition is that ***node.js is a JavaScript runtime built on Google’s open source v8 JavaScript engine***. Now what does that actually mean!!

Well, let’s start by trying to understand what a **JavaScript runtime** and the **v8 engine** actually are. So you have probably already used JavaScript before and it was probably always just inside a browser because any browser natively understands html, css and JavaScript and no matter if you write vanilla JavaScript or some JavaScript framework like react or angular code, that’s all just JavaScript that gets executed right in the browser. So in this case ***the browser is then the JavaScript runtime***.

But what if we could take JavaScript out of the browser and simply execute our JavaScript code somewhere else without all the restrictions that we have in the browser?

Well it turns out that we actually can and the solution for this as you can guess is called node.js and so node.js is just another JavaScript runtime.

It’s just like a container like an environment in which a program written in JavaScript can be executed but outside of any browser.

It’s actually a bit more complex than this of course but for now, this is enough.

*Who actually does execute the code if not the browser?*

And that’s where the v8 engine developed by Google comes into play because that is exactly where JavaScript code will be parsed and run in node.js.

So I hope that now the definition of node.js being a JavaScript runtime built on the v8 JavaScript engine makes a lot more sense and again this is just a very high level overview of what node is and how it works.

We are going to go into a lot of detail about how node really works behind the scenes in future but at this point I just want you to get a quick overview so that you can start learning the fundamentals of node.js in practice.

Now that we have JavaScript outside of the browser in a kind of standalone environment, which is node.js, we can do so many more things with JavaScript that were completely impossible before like accessing the file system or better networking capabilities are now possible with node.js and all these factors together give us the perfect conditions for using node.js as a web server, meaning that now we can finally use JavaScript on the server side of web development in order to build fast highly scalable network applications for powering the back end of our websites or web applications and this is absolutely fantastic and game changing for web development.

Let’s now take a look at some use cases for node and why it’s such a great fit for back-end development. The first thing that we need to talk about is the fact that node applications are so fast and so scalable because *node.js is single threaded and based on an event-driven non-blocking I/O model which makes node.js very lightweight and efficient*.

That, of course, sounds super complicated I know but once more we’re gonna talk about exactly what all of this means a bit later in the series but for now just keep in mind that node is perfect for building super fast and scalable data-intensive web applications and that makes node.js a perfect fit for building all different kinds of applications like building an API with a database behind it and preferably a non-relational NoSQL database like mongodb and this is actually exactly what we’re going to do later in this series as we dive deeper and deeper into node.js but there is of course all sorts of other apps that we can build like data streaming applications such as youtube or netflix, real-time chat apps or even server-side web applications where the entire content is simply generated right on the server.

So as you see the possibilities really are endless but there is also a type of apps that we should not build with node and this is when our app needs some super heavy server side processing like heavy image manipulations, video conversion, file compression or anything like that and you will learn why that is once we dive deeper into how node actually works. In this case we’re better off using something like ruby on rails, php or python because node.js just wasn’t developed for these use cases and although there are solutions for dealing with them we should always use the right tool for the task at hand.

So these are the types of applications that we can and cannot build using node.js and if you choose to use node, well, then you are in good company because many top companies like netflix, uber, paypal, ebay and many many more have already started using node.js, which really proves that the technology is stable enough at this point to be used in production and of course there are more reasons for using node and the biggest one is probably that using the same language across your entire stack both on the front end and the back end is a lot easier for you or for your team.

First, you don’t have to learn a new programming language for building your backend and also you don’t have to keep switching back and forth between languages, which will make the entire building process faster and more efficient, believe me it’s really really amazing to be able to use JavaScript to build your entire application from start to finish.

Another great reason for using node is that there is a huge library of open source packages or modules as we can also call them available for everyone for free. It’s called **npm** and you’ve probably heard of it or even used it already.

Finally, it’s also good to know that there is a huge active community of node developers out there that are constantly improving node.js and helping beginners and so you can really rely on the technology for all your projects in the years to come.

So that was a very brief intro and overview of node.js and some of its use cases.

We will next use node to create a very simple web server and build an extremely simple API so that you can get familiar with the basic principles of node development.

**✅ TODO TIME:**

Install [node.js](https://nodejs.org/en/) in your computer. *(Download the****LTS****version not the****Current****one)*