**Introduction to Node.js**

Okay. I’ll start.

My name is Abasifreke Ekwere and I’m a javascript developer.

I also do PHP/Laravel, flutter and pretty much anything that interests me.

I will like this quick class to be as interactive as possible.

I love JavaScript.

I hate JavaScript

But the obvious question you have or have had since joining this channel is; What is node Js?

I could simply quote a google definition and drop here and well, move ahead but let me tell you a story

In the late 90’s, Brendan Eich invented JavaScript

Guess how long it took him? 10 days

Many argue that this is the reason you have quirky behaviors like ```2 == ‘2’``` evaluating to true where the first 2 is a number and the other 2 is a string.

in a language like java (which is no way related to JavaScript), that will throw an error.

Why is this the case? Java is a static-typed language. JavaScript is dynamic.

static typed ? dynamic? How does this fit in the context of a beginner ?

in a static typed language, you need to declare the type of a variable before even going on to do anything with it:

e.g  
  
Number char = 2

this simply says whatever you give to char (a variable) must be a number

if i did something like this; Number char = ‘2’;  
  
it would throw an error

here’s what a static typed language does.

you declare a variable say char

guys, think of variables as containers that hold information.

those information are values

so for an expression like   
  
num = 1  
  
num is a container that holds 1

so to retrieve and reuse 1, you simple call on its container num

it will always give you 1.

**@here** do you get this analogy? good.

java does this

you declare num = 3

it looks for the type declaration.   
  
here’s a type declaration; Number num

you explicitly state that it’s a number;

it does not find a type declaration so throws an error that you should declare a type

you declare num = 3

it’s simply saying; better tell me what you want your container to contain before hand so i can look out for that in an orderly manner

okay so you proceed to do this:

.

.

Number num = ‘2’;

java looks for a type declaration, gives you a thumbs up

tries to store the value in ’2’  
in that container

lol there’s a problem

what the problem ? **@here**

you can’t store a string in a number container

this is very important and helpful when you grow into programming

what will javascript do ?

you do: var num;

var is just a syntax you use to declare variables in JS

there are others like; let and const:

very helpful but have deep underlying reasons as to why you could use them

we will stick with var for now for the sake of simplicity

you could also do;  
  
num;

that will become a global variable;

in javascript, you have the concept of environments

on the browser, you have window

on the server, you have global

we, node js developers are mostly only concerned with global

this is just a side note: let’s go back to what javascript would do;

it will simply just set num to undefined and let your application go on gracefully

so if you tried to access num  
you get undefined;

what will happen if you did;  
  
var num = ‘2’; ?

do you get an error or not if you tried to access num? you don’t get an error

javascript looks at the value about to be stored in num and internally determines its type and stores it in its container num (edited)

does it gracefully without errors.

This is a very powerful feature of dynamic typed languages like javascript.

you can use it to your advantage or you can let it give you nightmares

i assume you all understood that concept well enough :slightly_smiling_face:

the difference between loosely typed (dynamic) and static typed languages.

okay i don’t want to bore you so much tonight. i just wanted to introduce an easy yet very important concept generally found across programming languages

In 2009, javascript was brought to the server side invented by Ryan Dahl

We call it Node JS.

it offers a way to communicate with a database, build a server that can communicate with a client and even more

when you visit HNG board and try to log in, there’s something that helps identify you

a server and a database

that server could be built with Node JS

when you click that log in button, the browser sends your details to the server

the server already has implementations set in place to check your details

your details are stored in a database like MySQL, Mongo DB etc

so we write scripts to check your details against a database to confirm it really is you and we let you in or just tell you you can’t go in

that’s simply what node js is in a nutshell.

:clap::skin-tone-5: :muscle: :ok_hand::skin-tone-5:

what resources would i recommend for you to learn node js as beginner ?

okay. how did i even learn node js ?

i wrote my first line of code on April, 2nd, 2019

it was a simple “Hello World”

i used an app called “solo learn”

a few months later, i joined the pioneer set of start ng

i had just meddled into javascript slightly

that’s where it all got serious

just like it is just getting serious for a number of you here

i had a burning passion and desire to learn so i literally wrote code everyday day and moment i had

**i started at** [**w3schools.com**](http://w3schools.com/) **officially.**

they have a very nice javascript section

also, [**freecodecamp.org**](http://freecodecamp.org/) **is worth your time too.**

they both have very nice node js sections.

here’s what you should know, you own your personal progress.

A tutor can give you tips, directions and impart knowledge as they can

but your progress is determined mostly by you

visit those two sites and start getting acquainted with the basics of node js

so you can be ahead of the class and also discover things you may not be told in class

very soon, we will be pouring out snippets and code.

i’m here for the beginners. if you’re an expert, sorry for painlessly watching me explain concepts you knew since 1960

here’s a resource to get you started: <https://m.youtube.com/watch?v=fBNz5xF-Kx4>

[YouTube](https://www.youtube.com/)[**YouTube**](https://www.youtube.com/) | [Traversy Media](https://www.youtube.com/user/TechGuyWeb" \t "_blank)

[**Node.js Crash Course**](https://m.youtube.com/watch?v=fBNz5xF-Kx4)

here’s a reference material:<https://www.w3schools.com/nodejs/>

it is important you go through the video especially and use the reference material to clear a blocker

your tasks will revolve around them :slightly_smiling_face:

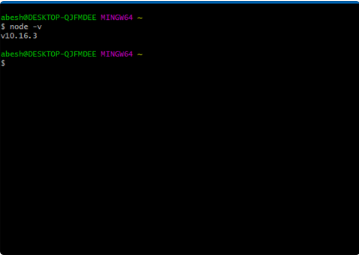
Before we get started, does everyone have node install on their computer ?

Open up your terminalSearch for terminal on your computer, make sure it is openand type node -v

Today we will be diving straight into code.

:+1:3

You should see something like this when you run that command



if you no not see that, visit here: <https://nodejs.org/en/download/>

download node js and type it afterwards..

you will see it

A simple challenge;

**Create a folder and name it after your slack username:**

**Create a file within that folder and name it: app.js**

**inside your app.js file, write console.log('My name is \*insert your name\* and I love Node JS')**

**open your terminal under the directory you are currently at,to make it easy, you should be using VScode.**

**in vscode, you can press CTRL + J to open/ close a terminal below your workspace**

**once your terminal is open, type node app.js**

I will move on

We will be creating a simple server.

Clear the console.log from your app.js file

make sure you have an empty file;

var http = require("http");http.createServer(function(req, res) {  
 res.writeHead(200, { "Content-Type": "text/plain" });  
 res.end("Hello World!");  
 })  
 .listen(8080);

Paste the above snippet in that empty app.js file.

Do not worry, i will go explain what it does :slightly_smiling_face:

run node app.js

visit <http://localhost:8080/> on your browser after running that

Ladies and gentlemen, you have successfully created a server and you should find that fun :grinning:

okay.what does it do ? how can i open that on a browser and see what I just wrote on my editor ?

is it magic ?

oluwa wetin dey occur ?

Think no further.

Yesterday I talked about environments

i presume your server is still running

type CTRL + C on your terminal Make sure to click on the terminal first

that terminates the server

okay do this:

type node

it takes you to the next line with a symbol like this: >

**@here** do you see that ?

excellent.

type console.log(global)

awesome.  
do this next;  
console.log(http)

okay great

the http module is part of the global environment

to use it, you need to import it to your desired file location

there are two major ways to import files and modules in javascript

1. The ES6 way

example:

import http from 'http'

2) The ES5 way

var http = require('http');

The first one is called an ES6 import

The second is called the common JS import

The second method is native to the node global environment and is heavily recommended

The first mostly requires transpiling by a tool like babel to be better understood by systems and browsers

so as node js developers, we stick to the second way of importing.

**@here** do you follow ?

http is an object found inside the global object

in javascript, you can nest objects within objects and within objects

lemme visualize this object for you

var global = {  
 http : {  
 createServer: function(){}  
 listen: function(){}  
}  
}

do you understand what is going on ? **@here**

global is an object that has properties and methods

var intern = {  
 track: "Backend", // property.  
 submitTask: function() {  
 console.log('just submitted my task')  
} // method.  
}

in the context of above example, what would you consider http, createServer and listen ?A method or a property

so let us proceed

http. createServer() calls the createServer method

the create server does what it implies it creates a server

when we move to express in the long run, you will see a very elegant way of doing this

not the express that egungun enters o

Express JS - The Node JS Framework

then we listen on a port listen(8080)

you can listen on any port of your choice but it is nice to use ports like 3000, 8000, 8080

makes more sense sematically imagine visiting localhost:1000

so we set headers

var http = require('http');http.createServer(function (req, res) {  
 res.writeHead(200, {'Content-Type': 'text/plain'});  
 res.end('Hello World!');  
}).listen(8080);

the createServer method takes in a callback

**what is a callback ?**

In javascript, functions are first class citizens

Just keep that at the back if your mind

I will explain that much much later

So you can pass functions as an argument

just like we pass a function to createServer

that function that is passed to createServer, takes in two arguments too req, res

req and res are objects

how do I know this ? it is part of the node js implementation

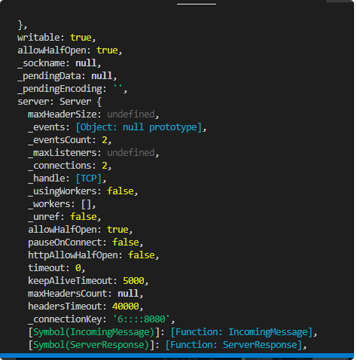
var http = require("http");http  
 .createServer(function(req, res) {  
 console.log(req, res);  
 res.writeHead(200, { "Content-Type": "text/plain" });  
 res.end("Hello World!");  
 })  
 .listen(8080);

paste the above guys and visit localhost:8080

then check your terminal.

a screenshot fo what you see in your terminal after visiting that link

these are the response and request objects



they do no make sense surely but they have methods and properties so they are objects

var http = require("http");http  
 .createServer(function(req, res) {  
 console.log(req, res);  
 res.writeHead(200, { "Content-Type": "text/plain" });  
 res.end("Hello World!");  
 })  
 .listen(8080);

so res.writeHead() is a method inside the response object

that method sets a header for what will be sent to the browser200 is an html status code.the content type here we say should be a plain text

we also need to end the session with res.end() so our browser does not keep spinning

do not believe ?

try this.

var http = require("http");http  
 .createServer(function(req, res) {  
 console.log(req, res);  
 res.writeHead(200, { "Content-Type": "text/plain" });  
 res.write("Hello World!");  
 })  
 .listen(8080);

notice that i used res.write('hello world')

do CTRL + C in your terminal

do node app.js again

visit localhost:8080

does it stop loading ?

try this

var http = require("http");http  
 .createServer(function(req, res) {  
 console.log(req, res);  
 res.writeHead(200, { "Content-Type": "text/plain" });  
 res.write("Hello World!");  
 res.end();  
 })  
 .listen(8080);

and follow the steps i highlighted

it will show Hello World this time

you can see res.end() (edited)

that terminates the session

we have successfully understood how to create a simple server

**@channel** I will be releasing your task shortly.

It will build upon what i just taught and will require you to do a bit of research.

it will be very simple and only needs you to google which is what developers do.

That will be all for now guys. We will be holding our classes regularly.  
Almost every day.  
At the end of this program, you will leave with a very good understanding of node JS.

you will also play around with express and write an API.