Node intro:

V8 engine - default engine inside the browser to run the java script application

java script is a single threaded environment, only one task can be done at a time. Eg. while the file upload is happening, no other task can be done. the page will freeze becasue of the single threaded environment.

in the java script environment asynchronous functions and callbakcs area also present. which we use to make it as a multi thread.

Javascript is an object based programming.

---------------------------------------------------------------------------------------------------------------------------------

Node JS –

once you install node JS you will be able to use node and npm(node packaging manager).

to run the node application: *node filename.js*

to debug the application: *node --inspect-brk filename.js*

to do the debugging open chrome and “*chrome://inspect*” you will see the application running over there

**nodemon**: it’s a seperate command, third party library which needs to be installed. This will help us in restarting the engine each time we make any changes

how to install any library?

*npm install library-name* or *npm install library-name@version*

*npm install library-name –g/-S/-D  
 (-g) -> global  
 (-S/-D) -> local  
 -S => dependency (depency required both in productio and development)  
 -D => development dependency (dependency required only in development not in production)*

setTimeout() – executes a timeout for whatever time mentioned in the function.

Define a variable – *let, constant* (var is completely out of picture in case of java script now)  
 To define a constant variable we will use *const* otherwise we will be using *let.*

IIFE – closure function. The function gets called automatically just after it is defined.

Destructuring syntax:

Callbacks:

Logical search fumction in any browser:

*Keypress -> ajax -> server -> response -> suggestion*

Callbacks and promises:

Modules:  
 each js file is a module

EventEmitter:

const baseUrl = 'http://localhost:3000';

const getAllEmployee = () => {

$get.JSON(`${baseUrl}/employee`).then(

data => {

console.log(data);

}

).catch(

err => console.log(err)

);

}

const userLogin = (user) => {

$.post(`${baseUrl}/login`,user).then(

data => {

console.log(data);

}

).catch(

err => console.log(err)

);

}

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Document</title>

</head>

<body>

<form>

<label>

Email

</label>

<input type="userEmail"/>

<label>

Password

</label>

<input type="userPassword"/>

<button>Submit</button>

</form>

<script src="node\_modules/jquery/dist/jquery.min.js"></script>

<script src="js/main.js"></script>

</body>

</html>