Date : 23rd - Nov- 2020
Morning Session : 9am – 11.00 PM
By ~ Akash

Topics: Intro to Node.js

What is Node.js?

Node.js is single thread non blocking async input out

Its js run time, it works on chrome v8 engine, its combination of c++ and javascript.

Node.js is most popular

1) it is fast -> why is it fast ? it is fast becoz of processing
2) Convert the code to machine language
3) Its do parallel processing
4) Do not block

Call Back: Callback is an asynchronous equivalent for a function. A callback function is called at the completion of a given task. Node makes heavy use of callbacks. All the APIs of Node are written in such a way that they support callbacks.

For example, a function to read a file may start reading a file and return the control to the execution environment immediately so that the next instruction can be executed. Once file I/O is complete, it will call the callback function while passing the callback function, the content of the file as a parameter. So there is no blocking or wait for File I/O. This makes Node.js highly scalable, as it can process a high number of requests without waiting for any function to return results.

REPL: REPL stands for Read Eval Print Loop and it represents a computer environment like a Windows console or Unix/Linux shell where a command is entered and the system responds with an output in an interactive mode. Node.js or Node comes bundled with a REPL environment. It performs the following tasks:

- Read - Reads user's input, parses the input into JavaScript data-structure, and stores in memory.

● Eval - Takes and evaluates the data structure.
● Print - Prints the result.
● Loop - Loops the above command until the user presses ctrl-c twice. The REPL feature of Node is very useful in experimenting with Node.js codes and to debug JavaScript codes.

NPM :

Node Package Manager (NPM) provides two main functionalities −

- Online repositories for node.js packages/modules which are searchable on search.nodejs.org
- Command line utility to install Node.js packages, do version management and dependency management of Node.js packages.

To verify the same, open console and type the following command

$ npm --version

Installing Modules using NPM

There is a simple syntax to install any Node.js module −

$ npm install <Module Name>

Global vs Local Installation

By default, NPM installs any dependency in the local mode. Here local mode refers to the package installation in node_modules directory lying in the folder where Node application is present. Locally deployed packages are accessible via require() method.

globally installed packages/dependencies are stored in system directory. Such dependencies can be used in CLI (Command Line Interface) function of any node.js but cannot be imported using require() in Node application directly.

JSON: JavaScript Object Notation (JSON) is a standard text-based format for representing structured data based on JavaScript object syntax. It is commonly used for transmitting data in web applications (e.g., sending some data from the server to the client, so it can be displayed on a web page, or vice versa)

Package.json :

package.json is present in the root directory of any Node application/module and is used to define the properties of a package.

```
///////////////
Step to Generate package.json
///////////////
*node js must be install in your system (node -v)
> navigate to the folder using cmd
> npm init
> answer all questions
> type "yes"
```

NPM Module:

```
///////////////
Step to install packages
///////////////
*node js must be install in your system (node -v)
> navigate to the folder using cmd
npm i packagename
npm i packagename --save

npm i packagename --save-dev
npm i -g packagename

//////
local >>>Local to folder
dev   >>> ist local to folder fro development not production
global >>> global to the system (to run /generate)
          Need Admin permission to install
          they will not add in package.json
```

```
////////////////
Step to install Global packages
////////////////

#window
> run your cmd as Admin
> npm i -g packagename


#mac/linux
> run terminal
> sudo npm i -g packagename
```

Inbuilt Packages:

Node.js - File System

Resource : [Node.js - File System - Tutorialspoint]