**EXPRESSJS**

**INTRODUCTION:**

Express java script is a open-source Node.js framework for quickly creating dynamic end-to-end **REST APIs.**

**SETUP:**

1. Install **node.js**
2. Create a folder myapp using **mkdir myapp**
3. Move to myapp folder using **cd myapp**
4. Create package.json using **npm init**
5. Install expressjs module using **npm install express –save**

**Simple GET API Hello World Example:**

1. **C**reate a file app.js inside **myapp** folder.
2. Write below snippet of code in **app.js** file.

****

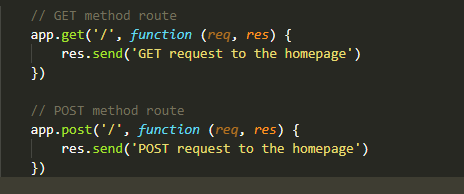
1. Run this file from **command prompt** using **node app.js**
2. Test Hello word example using URL **:** [**http://localhost:3000/**](http://localhost:3000/)

**Route methods:**

Express supports methods that correspond to all HTTP request methods.

A route method is derived from one of the HTTP methods,

The following code is an example of routes that are defined for the GET and the POST methods to the root of the app.



**MiddleWares:**

Simple Login middleware:

You’ll start by making your app log requests. Put the code in the following listing

inside **app.js.**

var express = require("express");

var app = express();

app.use(function(req, res, next) { //Logging Every income request

console.log("Request IP: " + req.url);

console.log("Request date: " + new Date());

next(); //It needs to call next to continue on to the next function in the middleware

//stack and needs to finish responding to the current incoming request Also.

});

app.get('/api', (req, res) => res.send('Hello World!'));

listen(3000, function() {

console.log("App started on port 3000");

});

DEAL WITH PATH PARAMS AND QUERY STRINGS IN URI:

Sample URI For Below snippet code: [**http://localhost:3000/**course/express](%20http://localhost:3000/course/express)

app.get('/course/:name', function(req, res) {

res.send('course name :' + req.params.name); //

});