**Basic syntax of expressJS**

* const express = require('express'); // Returns a function
* var app = express(); // app Can be pass to the createServer as this is the listener
* app.use((req,res,next)=>{ //Implementation of the middleware

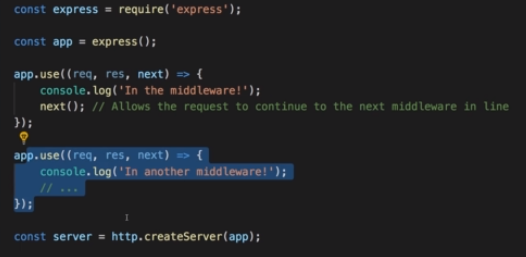
next();// Allows to go next

})

* var server = http.createServer(app);

**sending response from MIDDLEWARE**

* use res.send("<h1>HEADER</h1>"); // this will automatically sets the content type also and other header requirements as per the need

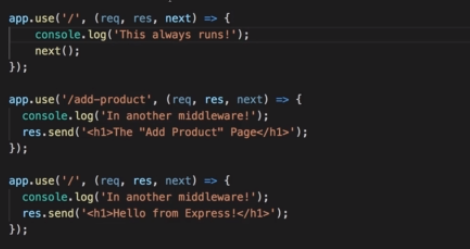


**Routing with ExpressJS**

-We can route using the **use()** function->mentioned above.

-A use function is overloaded with different type of arguments.

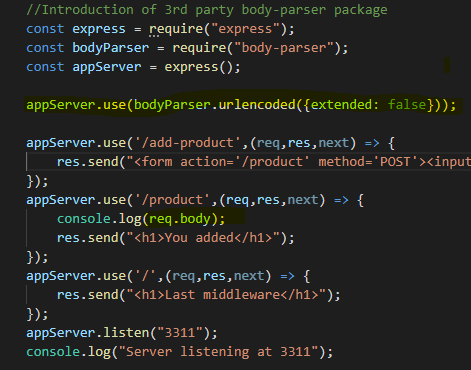
-We can provide 1st arg as path here(See below)



* Make sure that ‘/’ would be mentioned last as it will start with ‘/’ for any ‘/PATH’.Above next() is used.

**Parse requests using 3rd part tool Body Parser**

* See code 2\_parseRequests.js
* Saves use of buffer and handling of chunks and its conversion



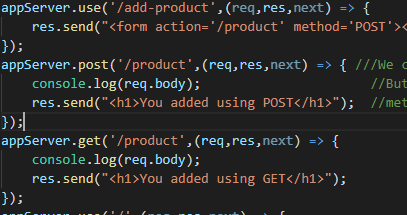
**USE GET POST**

- 3UseGetPost.js

**-**///We can use 'use' function for all html requests

/But we can also use POST and GTE to diff the

//method of the request

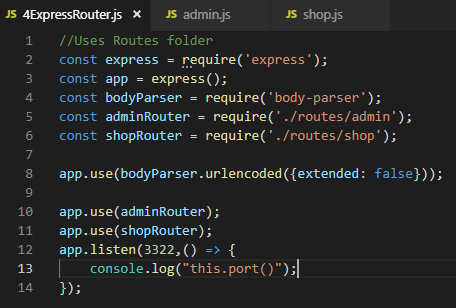
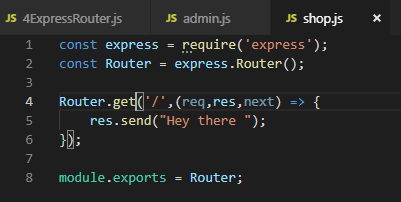
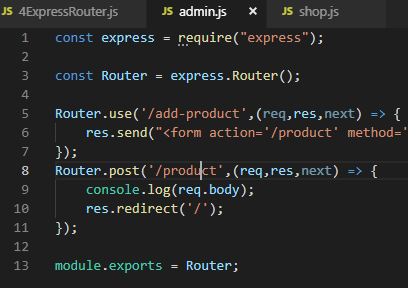


**Using Express Router**

**- 4ExpressRoter.js and Router Folder**

-We can put middleware code to another js file and can add this code as a module in main code.

-To do this express provides Router function that manages this other functions and by importing this router to main code we can use it as a handlebars.

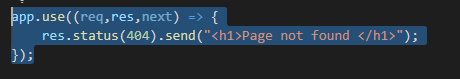


Adding 404 Error page

-After using get method (NOT ‘use’ ) for path ‘/’ this will definitely give below error if page not found.



-We can add below method that will handle any page request and display error to the user adding status as 404.



**Filtering Paths**

* 4ExpressRoter.js and Router Folder



* We can add a pre-path to the router that will be applied all request handlers inside the router hence if adminRouter has path /path1 then this will be accessible at /admin/path1

**Serving HTML pages and Use of Core module ‘path’**

\_\_dirname -> directory name of this file

‘../’ -> to go one step back as other folder is sibling

‘views’ -> sibling folder views containing htm files

‘filename.htm’ -> file to serve

const path = require('path');

const Router = express.Router();

Router.get('/',(req,res,next) => {

res.sendFile(path.join(\_\_dirname,'../','views','shop.htm'));

});

Using Helper function for Navigation