PROJECT 4 – MEAN Stack implementation in Ubuntu AWS

M = MongoDB (this is a document database that stores and allows to retrieve data

E = Express (This is a Back-end application framework that makes requests to database for reads and writes

A = **Angular** (This is a Front-end application framework that handles client and server requests)

N = Node.js (This is a JavaScript runtime environment that accepts requests and displays results to end user) i.e, *server-side JavaScript runtime environment*

TASKS

Pre-requisite

Create a new EC2 of Ubuntu t2.micro instance (name in Project_MEAN) on AWS and connect to terminal. Here, I made use of my windows powershell terminal.

Step 1. Installing Node.js – At this stage, Node.js is used to set up Express routes and AngularJs controllers.

Update Ubuntu

```
#sudo apt update
```

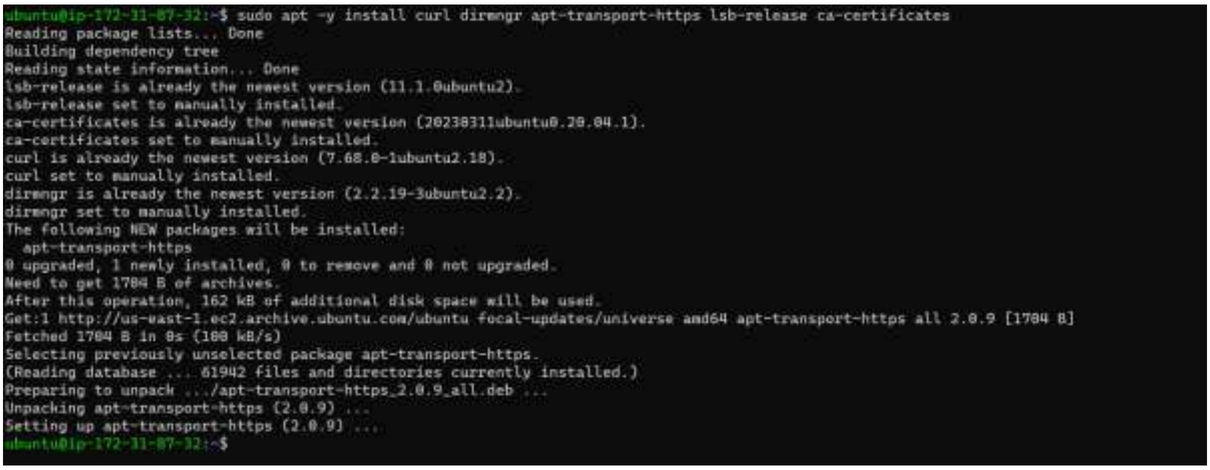
Upgrade Ubuntu

```
#sudo apt upgrade
```

Add certificates.

```
#sudo apt -y install curl dirmngr apt-transport-https lsb-release ca-certificates
```

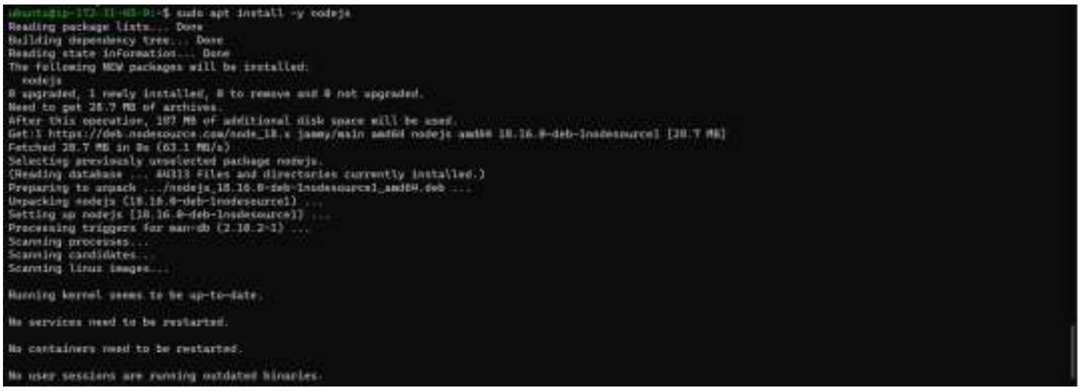
```
#curl -ls https://deb.nodesource.com/setup_18.x |sudo -E bash -
```



```
ubuntu@ip-172-31-87-32:~$ sudo apt -y install curl dirmngr apt-transport-https lsb-release ca-certificates
Reading package lists... Done
Building dependency tree
Reading state information... Done
lsb-release is already the newest version (11.1.0ubuntu2).
lsb-release set to manually installed.
ca-certificates is already the newest version (20230311ubuntu0.20.04.1).
ca-certificates set to manually installed.
curl is already the newest version (7.68.0-1ubuntu2.18).
curl set to manually installed.
dirmngr is already the newest version (2.2.19-3ubuntu2.2).
dirmngr set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 1704 B of archives.
After this operation, 162 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/universe amd64 apt-transport-https all 2.0.9 [1704 B]
Fetched 1704 B in 0s (100 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 61942 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.0.9_all.deb ...
Unpacking apt-transport-https (2.0.9) ...
Setting up apt-transport-https (2.0.9) ...
ubuntu@ip-172-31-87-32:~$
```

Install NodeJS

```
#sudo apt install -y nodejs
```



```
ubuntu@ip-172-31-40-9:~$ sudo apt install -y nodejs
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  nodejs
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 28.7 MB of archives.
After this operation, 187 MB of additional disk space will be used.
Get:1 https://deb.nodesource.com/node_18.x jammy/main amd64 nodejs amd64 18.16.0-deb-1nodesource1 [28.7 MB]
Fetched 28.7 MB in 0s (63.1 MB/s)
Selecting previously unselected package nodejs.
(Reading database ... 64313 files and directories currently installed.)
Preparing to unpack .../nodejs_18.16.0-deb-1nodesource1_amd64.deb ...
Unpacking nodejs (18.16.0-deb-1nodesource1) ...
Setting up nodejs (18.16.0-deb-1nodesource1) ...
Processing triggers for man-db (2.10.2-1) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.
```

Step 2:

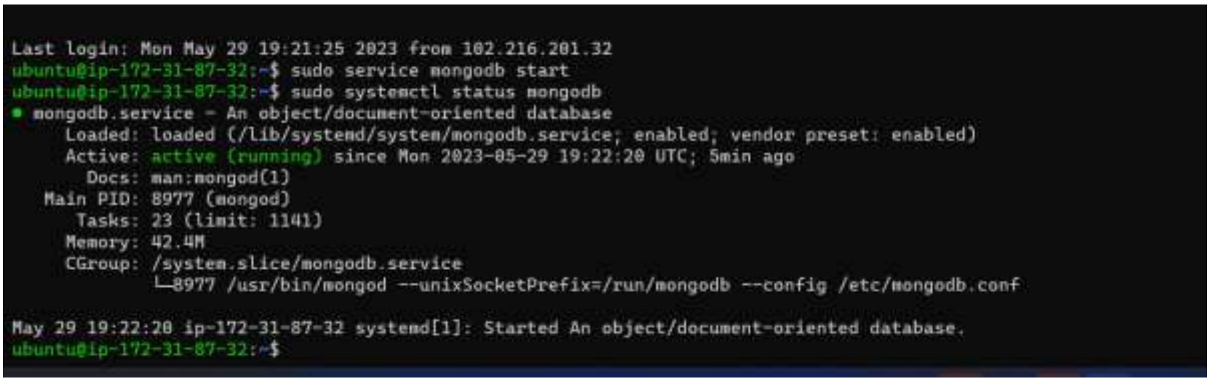
Installing MongoDB

```
#sudo apt install -y mongodb
```

Here, I faced an Error (Package 'Mongodb' has no installation candidate), this was because I was working on Ubuntu version 22.04. to resolve this error, I simply terminated the ubuntu 22.04 instance and launched a 20.04 instance which worked just perfectly.

Start the server and confirm if the server is up and running.

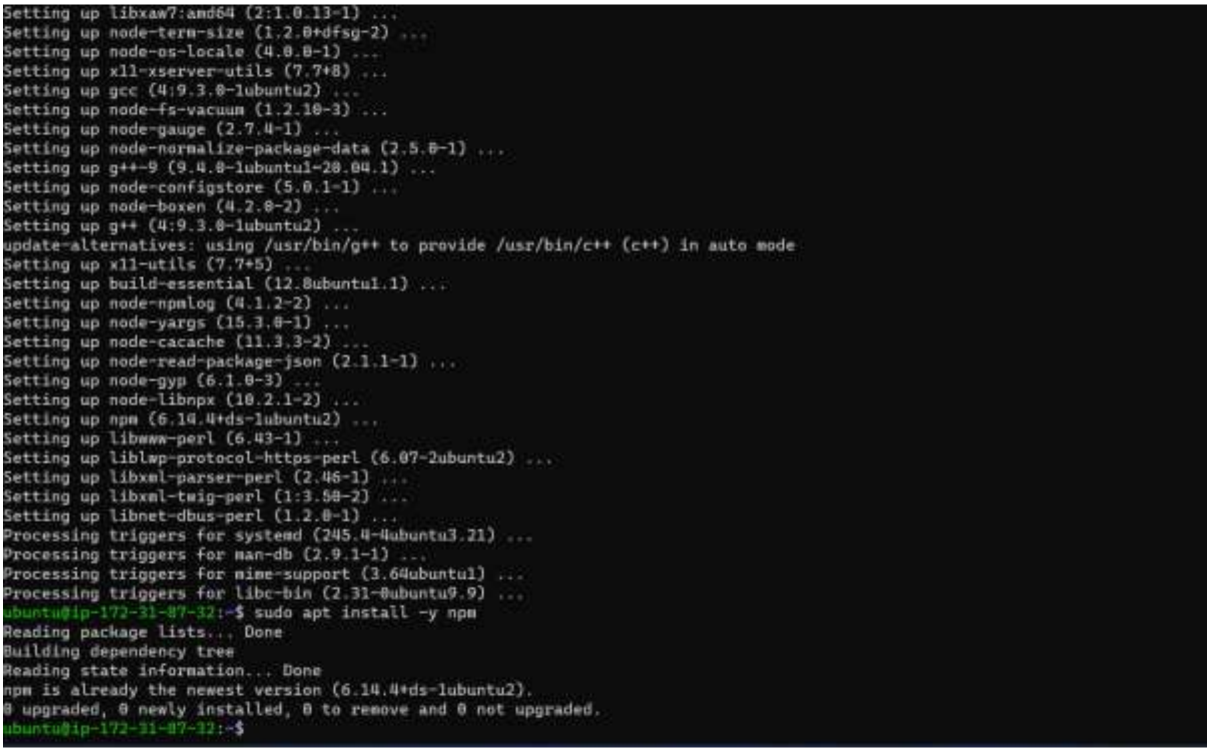
```
#Sudo service mongodb start && sudo systemctl status mongodb
```



```
Last login: Mon May 29 19:21:25 2023 from 102.216.201.32
ubuntu@ip-172-31-87-32:~$ sudo service mongodb start
ubuntu@ip-172-31-87-32:~$ sudo systemctl status mongodb
● mongodb.service - An object/document-oriented database
     Loaded: loaded (/lib/systemd/system/mongodb.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2023-05-29 19:22:20 UTC; 5min ago
       Docs: man:mongod(1)
   Main PID: 8977 (mongod)
      Tasks: 23 (limit: 1141)
     Memory: 42.4M
     CGroup: /system.slice/mongodb.service
             └─8977 /usr/bin/mongod --unixSocketPrefix=/run/mongodb --config /etc/mongodb.conf

May 29 19:22:20 ip-172-31-87-32 systemd[1]: Started An object/document-oriented database.
ubuntu@ip-172-31-87-32:~$
```

Now we have to install a node package manager NPM

```
sudo apt install -y npm
```



```
Setting up libxaw7:amd64 (2:1.0.13-1) ...
Setting up node-term-size (1.2.0+dfsg-2) ...
Setting up node-os-locale (4.0.0-1) ...
Setting up x11-xserver-utils (7.7+8) ...
Setting up gcc (4:9.3.0-1ubuntu2) ...
Setting up node-fs-vacuum (1.2.10-3) ...
Setting up node-gauge (2.7.4-1) ...
Setting up node-normalize-package-data (2.5.0-1) ...
Setting up g++-9 (9.4.0-1ubuntu1~20.04.1) ...
Setting up node-configstore (5.0.1-1) ...
Setting up node-boxen (4.2.0-2) ...
Setting up g++ (4:9.3.0-1ubuntu2) ...
update-alternatives: using /usr/bin/g++ to provide /usr/bin/c++ (c++) in auto mode
Setting up x11-utils (7.7+5) ...
Setting up build-essential (12.8ubuntu1.1) ...
Setting up node-npmlog (4.1.2-2) ...
Setting up node-yargs (15.3.0-1) ...
Setting up node-cacache (11.3.3-2) ...
Setting up node-read-package-json (2.1.1-1) ...
Setting up node-gyp (6.1.0-3) ...
Setting up node-libnpx (10.2.1-2) ...
Setting up npm (6.14.4+ds-1ubuntu2) ...
Setting up libwww-perl (6.43-1) ...
Setting up liblwp-protocol-https-perl (6.07-2ubuntu2) ...
Setting up libxml-parser-perl (2.46-1) ...
Setting up libxml-twig-perl (1:3.50-2) ...
Setting up libnet-dbus-perl (1.2.0-1) ...
Processing triggers for systemd (245.4-4ubuntu3.21) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.9) ...
ubuntu@ip-172-31-87-32:~$ sudo apt install -y npm
Reading package lists... Done
Building dependency tree
Reading state information... Done
npm is already the newest version (6.14.4+ds-1ubuntu2).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-87-32:~$
```

Install ‘body-Parser’

We need ‘body-parser’ package to help us process JSON files passed in requests to the server.

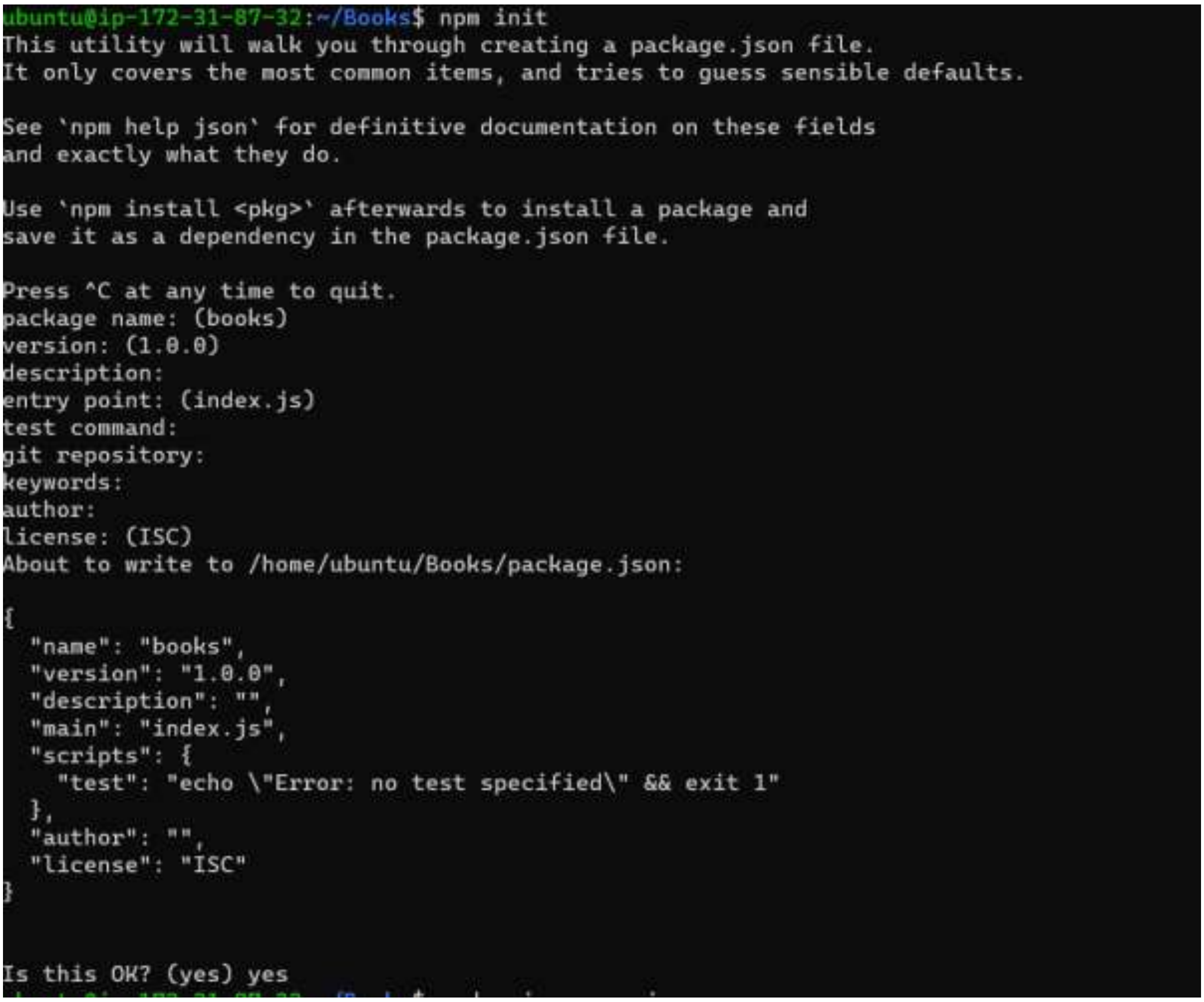
```
#sudo npm install body-parser
```

Create a directory called ‘Books’ and cd into the directory.

```
#mkdir Books && cd Books
```

Initialize the node package manager in the ‘Books’ directory.

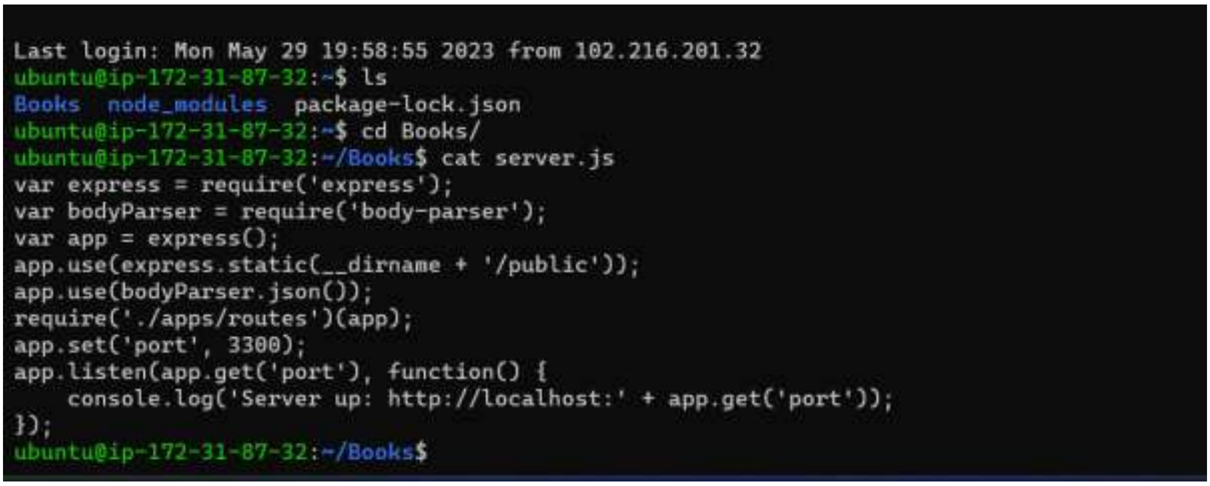
```
#npm init
```



```
ubuntu@ip-172-31-87-32:~/Books$ npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help json` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (books)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to /home/ubuntu/Books/package.json:

{
  "name": "books",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}


Is this OK? (yes) yes
```

In the Server.js file, past in the below code block

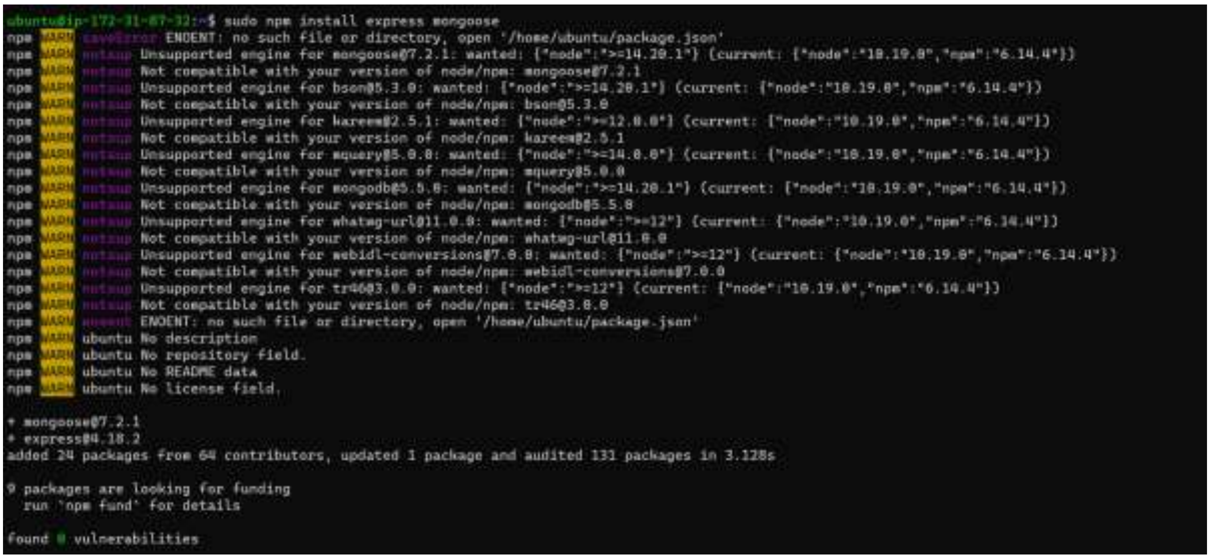
```
var express = require('express');
var bodyParser = require('body-parser');
var app = express();
app.use(express.static(__dirname + '/public'));
app.use(bodyParser.json());
require('./apps/routes')(app);
app.set('port', 3300);
app.listen(app.get('port'), function() {
    console.log('Server up: http://localhost:' +
app.get('port'));
});
```



```
Last login: Mon May 29 19:58:55 2023 from 102.216.201.32
ubuntu@ip-172-31-87-32:~$ ls
Books  node_modules  package-lock.json
ubuntu@ip-172-31-87-32:~$ cd Books/
ubuntu@ip-172-31-87-32:~/Books$ cat server.js
var express = require('express');
var bodyParser = require('body-parser');
var app = express();
app.use(express.static(__dirname + '/public'));
app.use(bodyParser.json());
require('./apps/routes')(app);
app.set('port', 3300);
app.listen(app.get('port'), function() {
    console.log('Server up: http://localhost:' + app.get('port'));
});
ubuntu@ip-172-31-87-32:~/Books$
```

Step 3: Installing Express and setting up routes to the Server

Step 3: Install Express.js and install mongoose



```
ubuntu@ip-172-31-87-32:~$ sudo npm install express mongoose
npm WARN saveError ENOENT: no such file or directory, open '/home/ubuntu/package.json'
npm WARN notsup Unsupported engine for mongoose@7.2.1: wanted: {"node":">=14.20.1"} (current: {"node":"10.19.0","npm":"6.14.4"})
npm WARN notsup Not compatible with your version of node/npm: mongoose@7.2.1
npm WARN notsup Unsupported engine for bson@5.3.0: wanted: {"node":">=14.20.1"} (current: {"node":"10.19.0","npm":"6.14.4"})
npm WARN notsup Not compatible with your version of node/npm: bson@5.3.0
npm WARN notsup Unsupported engine for kareem@2.5.1: wanted: {"node":">=12.0.0"} (current: {"node":"10.19.0","npm":"6.14.4"})
npm WARN notsup Not compatible with your version of node/npm: kareem@2.5.1
npm WARN notsup Unsupported engine for mquery@5.0.0: wanted: {"node":">=14.0.0"} (current: {"node":"10.19.0","npm":"6.14.4"})
npm WARN notsup Not compatible with your version of node/npm: mquery@5.0.0
npm WARN notsup Unsupported engine for mongodb@5.5.0: wanted: {"node":">=14.20.1"} (current: {"node":"10.19.0","npm":"6.14.4"})
npm WARN notsup Not compatible with your version of node/npm: mongodb@5.5.0
npm WARN notsup Unsupported engine for whatwg-url@11.0.0: wanted: {"node":">=12"} (current: {"node":"10.19.0","npm":"6.14.4"})
npm WARN notsup Not compatible with your version of node/npm: whatwg-url@11.0.0
npm WARN notsup Unsupported engine for webidl-conversions@7.0.0: wanted: {"node":">=12"} (current: {"node":"10.19.0","npm":"6.14.4"})
npm WARN notsup Not compatible with your version of node/npm: webidl-conversions@7.0.0
npm WARN notsup Unsupported engine for tr46@3.0.0: wanted: {"node":">=12"} (current: {"node":"10.19.0","npm":"6.14.4"})
npm WARN notsup Not compatible with your version of node/npm: tr46@3.0.0
npm WARN enoent ENOENT: no such file or directory, open '/home/ubuntu/package.json'
npm WARN ubuntu No description
npm WARN ubuntu No repository field.
npm WARN ubuntu No README data
npm WARN ubuntu No license field.

+ mongoose@7.2.1
+ express@4.18.2
added 24 packages from 64 contributors, updated 1 package and audited 131 packages in 3.128s

9 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

In the ‘Books’ directory, create a folder named ‘apps’ and cd in to the app directory

`#mkdir apps && cd apps`

Create a file named routes.js

`#sudo vi routes.js`

Paste the below code block into routes.js

```
const Book = require('./models/book');

module.exports = function(app){
  app.get('/book', function(req, res){
    Book.find({}).then(result => {
      res.json(result);
    }).catch(err => {
      console.error(err);
      res.status(500).send('An error occurred while retrieving books');
    });
  });

  app.post('/book', function(req, res){
    const book = new Book({
      name: req.body.name,
      isbn: req.body.isbn,
      author: req.body.author,
      pages: req.body.pages
    });
    book.save().then(result => {
      res.json({
        message: "Successfully added book",
        book: result
      });
    }).catch(err => {
      console.error(err);
      res.status(500).send('An error occurred while saving the book');
    });
  });
```

```
 app.delete("/book/:isbn", function(req, res){
  Book.findOneAndRemove(req.query).then(result => {
   res.json({
    message: "Successfully deleted the book",
    book: result
   });
  }).catch(err => {
   console.error(err);
   res.status(500).send('An error occurred while deleting the book');
  });
 });

 const path = require('path');
 app.get('*', function(req, res){
  res.sendFile(path.join(__dirname, 'public', 'index.html'));
 });
};
```

In the ‘apps’ folder, create a directory names ‘models’ and cd into ‘models folder

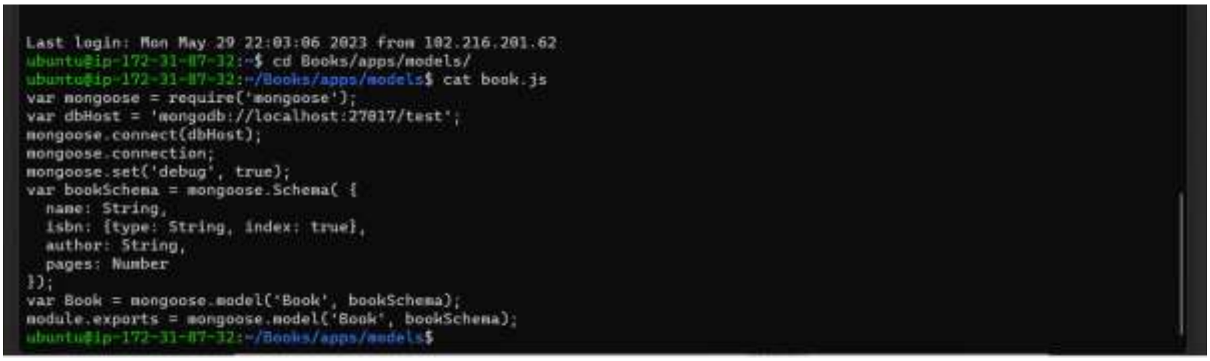
#mkdir model && cd models

Create a book.js file in the models directory.

#sudo vi models.js

Paste the below code block into models.js

```
var mongoose = require('mongoose');
var dbHost = 'mongodb://localhost:27017/test';
mongoose.connect(dbHost);
mongoose.connection;
mongoose.set('debug', true);
var bookSchema = mongoose.Schema( {
  name: String,
  isbn: {type: String, index: true},
  author: String,
  pages: Number
});
var Book = mongoose.model('Book', bookSchema);
module.exports = mongoose.model('Book', bookSchema);
```



```
Last login: Mon May 29 22:03:06 2023 from 102.216.201.62
ubuntu@ip-172-31-87-32:~$ cd Books/apps/models/
ubuntu@ip-172-31-87-32:~/Books/apps/models$ cat book.js
var mongoose = require('mongoose');
var dbHost = 'mongodb://localhost:27017/test';
mongoose.connect(dbHost);
mongoose.connection;
mongoose.set('debug', true);
var bookSchema = mongoose.Schema( {
  name: String,
  isbn: {type: String, index: true},
  author: String,
  pages: Number
});
var Book = mongoose.model('Book', bookSchema);
module.exports = mongoose.model('Book', bookSchema);
ubuntu@ip-172-31-87-32:~/Books/apps/models$
```

STEP 4 -Access the routes with AngularJS

This is required in other to connect our webpage with Express and perform actions on or book register.

To do this, change directory back into 'Books'.

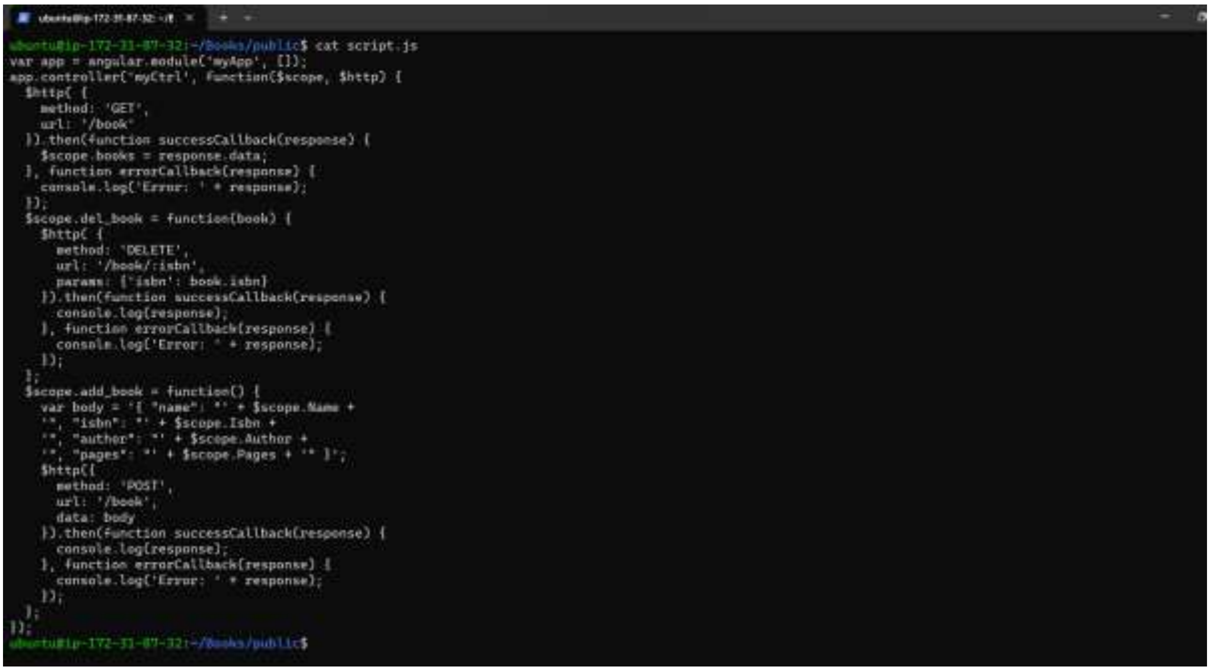
In the 'Books' folder, create a folder named 'public' and create a file in the 'public' folder named script.js

#mkdir public && cd public

#sudo vi script.js

`Paste the code block below into script.js`

```
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope, $http) {
  $http( {
    method: 'GET',
    url: '/book'
  }).then(function successCallback(response) {
    $scope.books = response.data;
  }, function errorCallback(response) {
    console.log('Error: ' + response);
  });
  $scope.del_book = function(book) {
    $http( {
      method: 'DELETE',
      url: '/book/:isbn',
      params: {'isbn': book.isbn}
    }).then(function successCallback(response) {
      console.log(response);
    }, function errorCallback(response) {
      console.log('Error: ' + response);
    });
  };
  $scope.add_book = function() {
    var body = '{ "name": "' + $scope.Name +
    '", "isbn": "' + $scope.Isbn +
    '", "author": "' + $scope.Author +
    '", "pages": "' + $scope.Pages + '" }';
    $http({
      method: 'POST',
      url: '/book',
      data: body
    }).then(function successCallback(response) {
      console.log(response);
    }, function errorCallback(response) {
      console.log('Error: ' + response);
    });
  };
});
```



```
ubuntu@ip-172-31-87-32:~/Books/public$ cat script.js
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope, $http) {
  $http( {
    method: 'GET',
    url: '/book'
  }).then(function successCallback(response) {
    $scope.books = response.data;
  }, function errorCallback(response) {
    console.log('Error: ' + response);
  });
  $scope.del_book = function(book) {
    $http( {
      method: 'DELETE',
      url: '/book/:isbn',
      params: {'isbn': book.isbn}
    }).then(function successCallback(response) {
      console.log(response);
    }, function errorCallback(response) {
      console.log('Error: ' + response);
    });
  };
  $scope.add_book = function() {
    var body = '{ "name": "' + $scope.Name +
    '", "isbn": "' + $scope.Isbn +
    '", "author": "' + $scope.Author +
    '", "pages": "' + $scope.Pages + '" }';
    $http({
      method: 'POST',
      url: '/book',
      data: body
    }).then(function successCallback(response) {
      console.log(response);
    }, function errorCallback(response) {
      console.log('Error: ' + response);
    });
  };
});
ubuntu@ip-172-31-87-32:~/Books/public$
```

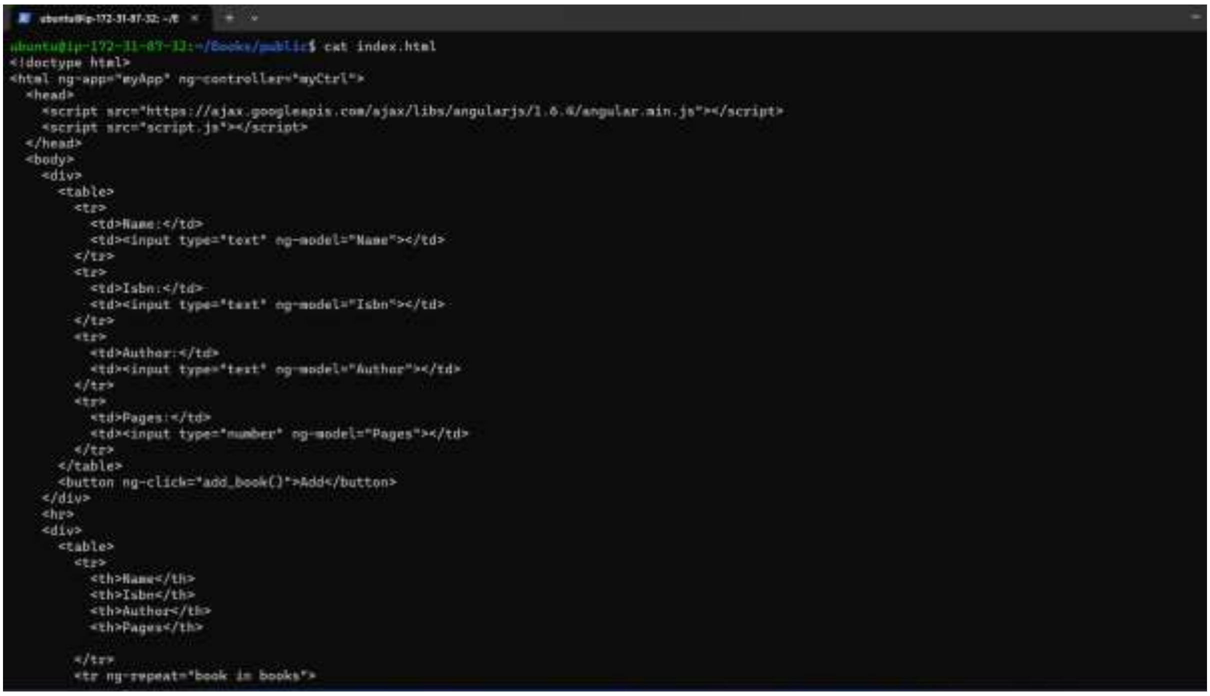
Still in the public folder, create a file named index.html;

#sudo vi index.html

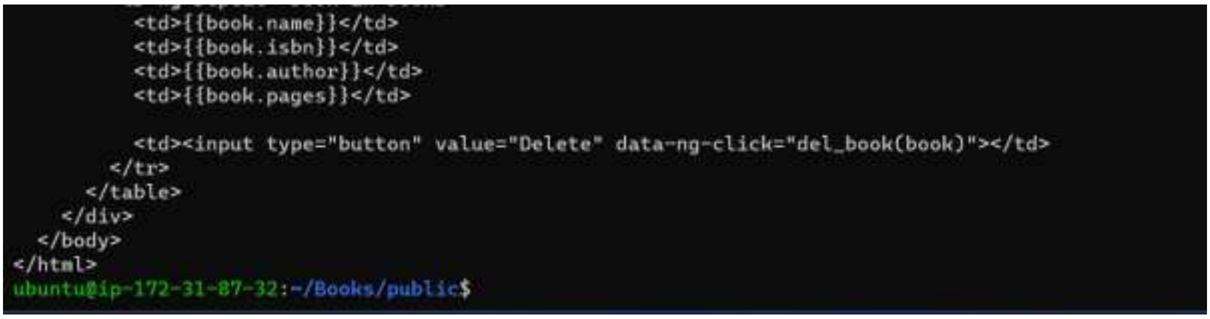
Paste the below code block into index.html file.

```
<!doctype html>
<html ng-app="myApp" ng-controller="myCtrl">
  <head>
    <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.4/angular.min.js"></
script>
    <script src="script.js"></script>
  </head>
  <body>
    <div>
      <table>
        <tr>
```

```
          <td>Name:</td>
          <td><input type="text" ng-model="Name"></td>
        </tr>
        <tr>
          <td>Isbn:</td>
          <td><input type="text" ng-model="Isbn"></td>
        </tr>
        <tr>
          <td>Author:</td>
          <td><input type="text" ng-model="Author"></td>
        </tr>
        <tr>
          <td>Pages:</td>
          <td><input type="number" ng-model="Pages"></td>
        </tr>
      </table>
      <button ng-click="add_book()">Add</button>
    </div>
    <hr>
    <div>
      <table>
        <tr>
          <th>Name</th>
          <th>Isbn</th>
          <th>Author</th>
          <th>Pages</th>

        </tr>
        <tr ng-repeat="book in books">
          <td>{{book.name}}</td>
          <td>{{book.isbn}}</td>
          <td>{{book.author}}</td>
          <td>{{book.pages}}</td>

          <td><input type="button" value="Delete" data-ng-click="del_book(book)"></td>
        </tr>
      </table>
    </div>
  </body>
</html>
```



```
ubuntu@ip-172-31-87-32:~/Books/public$ cat index.html
<!doctype html>
<html ng-app="myApp" ng-controller="myCtrl">
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.4/angular.min.js"></script>
    <script src="script.js"></script>
  </head>
  <body>
    <div>
      <table>
        <tr>
          <td>Name:</td>
          <td><input type="text" ng-model="Name"></td>
        </tr>
        <tr>
          <td>Isbn:</td>
          <td><input type="text" ng-model="Isbn"></td>
        </tr>
        <tr>
          <td>Author:</td>
          <td><input type="text" ng-model="Author"></td>
        </tr>
        <tr>
          <td>Pages:</td>
          <td><input type="number" ng-model="Pages"></td>
        </tr>
      </table>
      <button ng-click="add_book()">Add</button>
    </div>
    <hr>
    <div>
      <table>
        <tr>
          <th>Name</th>
          <th>Isbn</th>
          <th>Author</th>
          <th>Pages</th>

        </tr>
        <tr ng-repeat="book in books">
```



```
          <td>{{book.name}}</td>
          <td>{{book.isbn}}</td>
          <td>{{book.author}}</td>
          <td>{{book.pages}}</td>

          <td><input type="button" value="Delete" data-ng-click="del_book(book)"></td>
        </tr>
      </table>
    </div>
  </body>
</html>
ubuntu@ip-172-31-87-32:~/Books/public$
```

Cd into the ‘Books’ directory and Start the server by running the command below.

```
#node server.js
```

At this point, I received back an error below:

/home/ubuntu/node_modules/mongodb/lib/operations/add_user.js:16

```
    this.options = options ?? {};
                           ^
```

SyntaxError: Unexpected token ?

Which means my server did not start up.

To resolve this error, I found out that the node.js version was a bit old and not compatible with the nullish coalescing operator (??) {}. So I had to install the node version manager (nvm) for node.js in order to be able to update the existing node.js version v10.19.0 to version v18.16.0.

Run the script below:

#curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.3/install.sh | bash

Install latest node version manager (nvm)

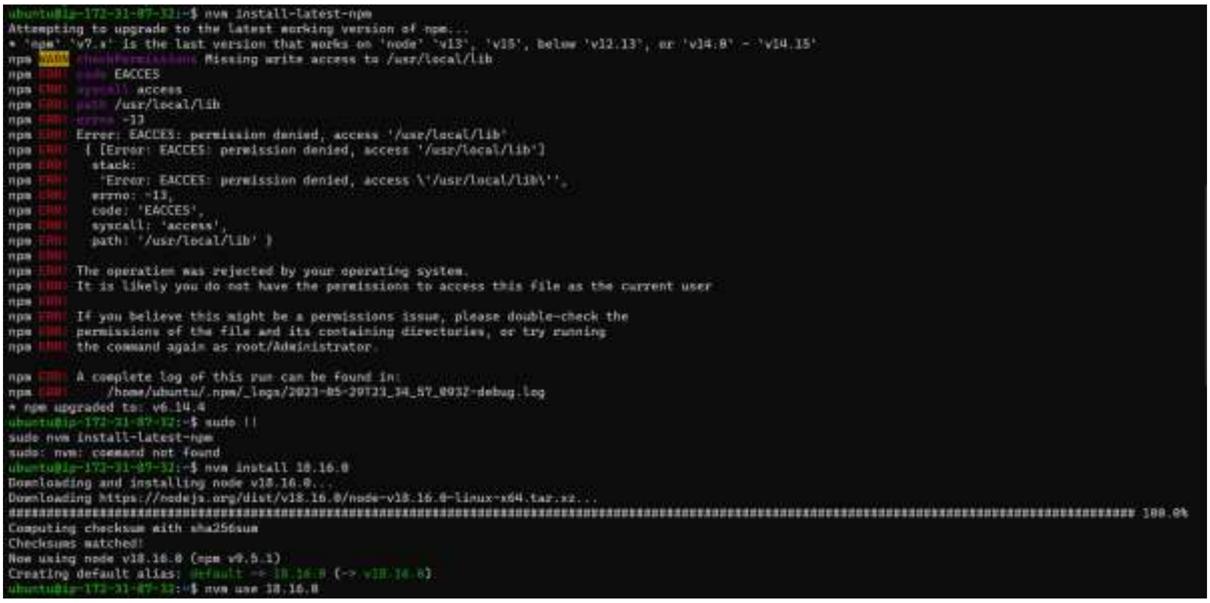
#nvm install-latest-npm

Once the NVM was installed, I closed my terminal, opened a new terminal, connected my instance before running the command below.

#nvm install 18.16.0

Set Node.js version 18.16.0 as the default version for your terminal session by running the following command:

#nvm use 18.16.0

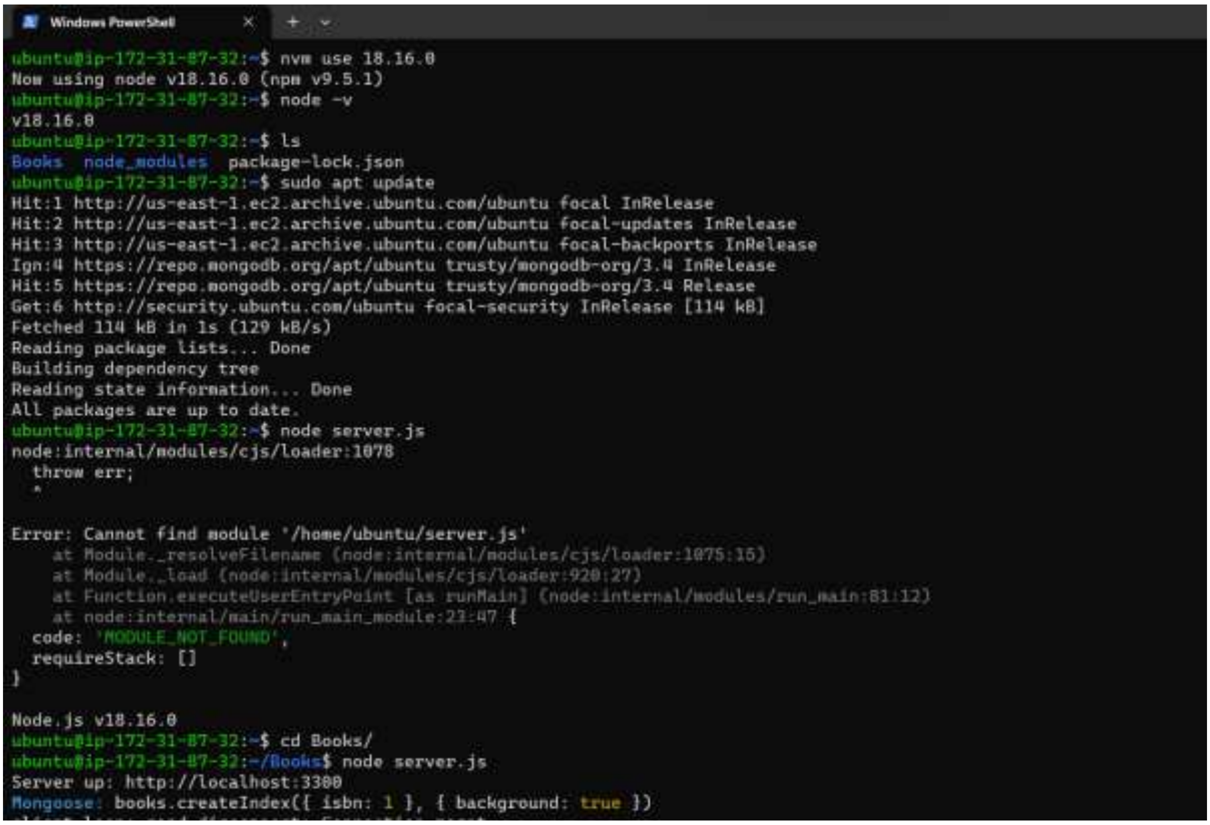


```
ubuntu@ip-172-31-87-32:~$ nvm install-latest-npm
Attempting to upgrade to the latest working version of npm...
* 'npm' 'v7.x' is the last version that works on 'node' 'v13', 'v15', below 'v12.13', or 'v14.0' - 'v14.15'
npm WARN checkPermissions Missing write access to /usr/local/lib
npm ERR! code EACCES
npm ERR! syscall access
npm ERR! path /usr/local/lib
npm ERR! errno -13
npm ERR! Error: EACCES: permission denied, access '/usr/local/lib'
npm ERR!  { [Error: EACCES: permission denied, access '/usr/local/lib']
npm ERR!   stack:
npm ERR!    'Error: EACCES: permission denied, access \'/usr/local/lib\'',
npm ERR!   errno: -13,
npm ERR!   code: 'EACCES',
npm ERR!   syscall: 'access',
npm ERR!   path: '/usr/local/lib' }
npm ERR!
npm ERR! The operation was rejected by your operating system.
npm ERR! It is likely you do not have the permissions to access this file as the current user
npm ERR!
npm ERR! If you believe this might be a permissions issue, please double-check the
npm ERR! permissions of the file and its containing directories, or try running
npm ERR! the command again as root/Administrator.

npm ERR! A complete log of this run can be found in:
npm ERR!     /home/ubuntu/.npm/_logs/2023-05-29T23_34_57_093Z-debug.log
* npm upgraded to: v6.14.4
ubuntu@ip-172-31-87-32:~$ sudo !!
sudo nvm install-latest-npm
sudo: nvm: command not found
ubuntu@ip-172-31-87-32:~$ nvm install 18.16.0
Downloading and installing node v18.16.0...
Downloading https://nodejs.org/dist/v18.16.0/node-v18.16.0-linux-x64.tar.xz...
######################################################################################################################## 100.0%
Computing checksum with sha256sum
Checksums matched!
Now using node v18.16.0 (npm v9.5.1)
Creating default alias: default -> 18.16.0 (-> v18.16.0)
ubuntu@ip-172-31-87-32:~$ nvm use 18.16.0
```

At this point, my server was running on node.js version 18.16.0

Now, Update any other dependencies.

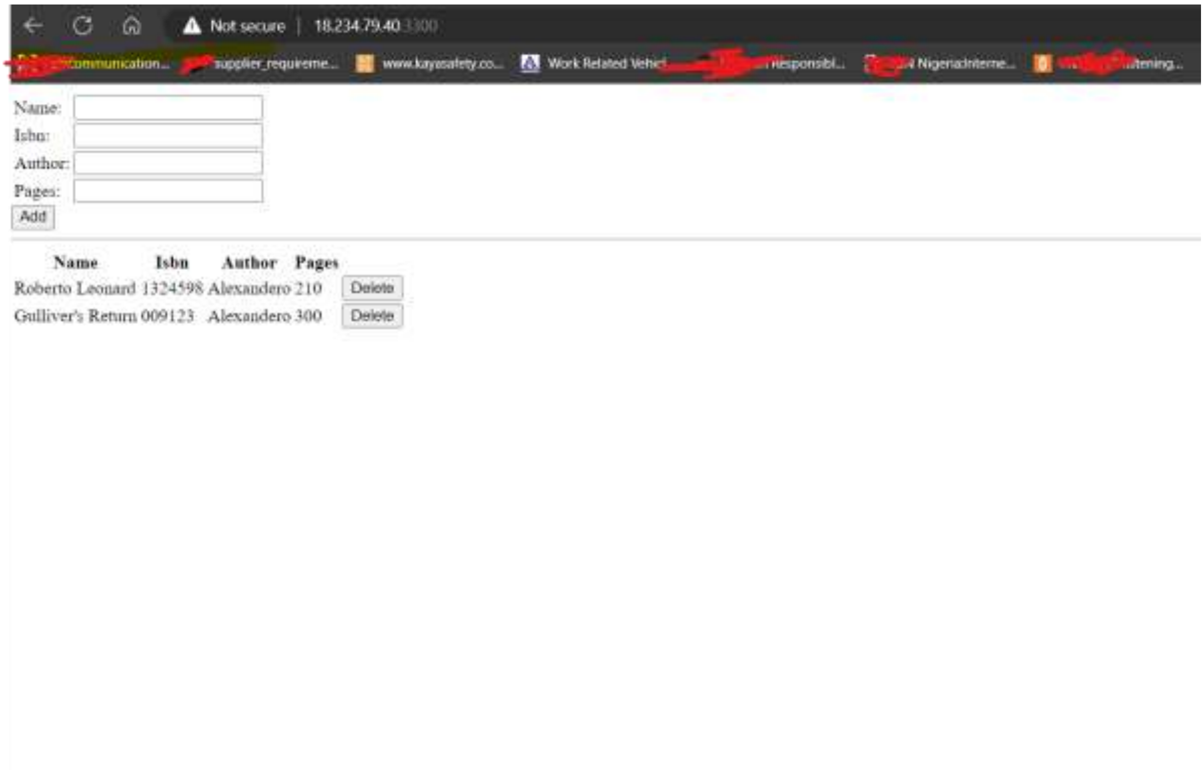
#sudo apt update



```
ubuntu@ip-172-31-87-32:~$ nvm use 18.16.0
Now using node v18.16.0 (npm v9.5.1)
ubuntu@ip-172-31-87-32:~$ node -v
v18.16.0
ubuntu@ip-172-31-87-32:~$ ls
Books  node_modules  package-lock.json
ubuntu@ip-172-31-87-32:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease
Ign:4 https://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.4 InRelease
Hit:5 https://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.4 Release
Get:6 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Fetched 114 kB in 1s (129 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
ubuntu@ip-172-31-87-32:~$ node server.js
node:internal/modules/cjs/loader:1078
  throw err;
  ^

Error: Cannot find module '/home/ubuntu/server.js'
    at Module._resolveFilename (node:internal/modules/cjs/loader:1075:15)
    at Module._load (node:internal/modules/cjs/loader:920:27)
    at Function.executeUserEntryPoint [as runMain] (node:internal/modules/run_main:81:12)
    at node:internal/main/run_main_module:23:47 {
  code: 'MODULE_NOT_FOUND',
  requireStack: []
}

Node.js v18.16.0
ubuntu@ip-172-31-87-32:~$ cd Books/
ubuntu@ip-172-31-87-32:~/Books$ node server.js
Server up: http://localhost:3300
Mongoose: books.createIndex({ isbn: 1 }, { background: true })
```

Now make sure port 3300 is opened on your security group and retest the server.

#node server.js



Not secure | 18.234.79.40:3300
[redacted] communication...
[redacted] supplier_requireme...
www.kayasafety.co...
Work Related Vehic[redacted] Responsibl...
[redacted] Nigeria Interne...
[redacted]tening...

Name:
Isbn:
Author:
Pages:

Add

Name	Isbn	Author	Pages	
Roberto Leonard	1324598	Alexandero	210	Delete
Gulliver's Return	009123	Alexandero	300	Delete