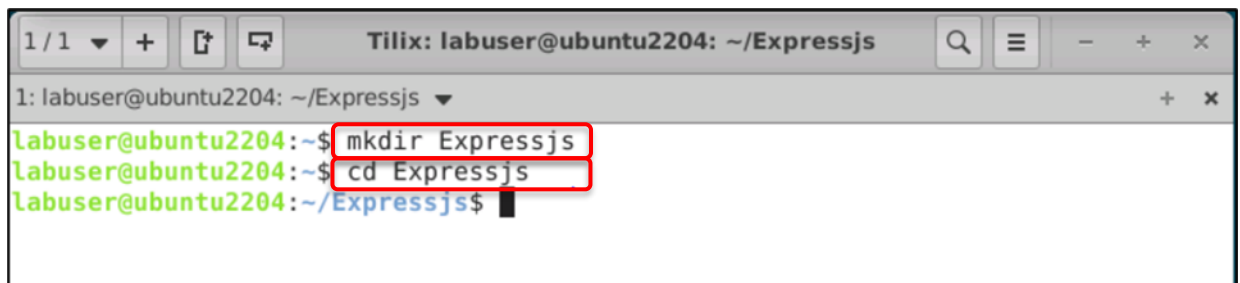




## Step 2: Install Express.js

- 2.1 Run the code below to create a directory to hold the application and make it the working:

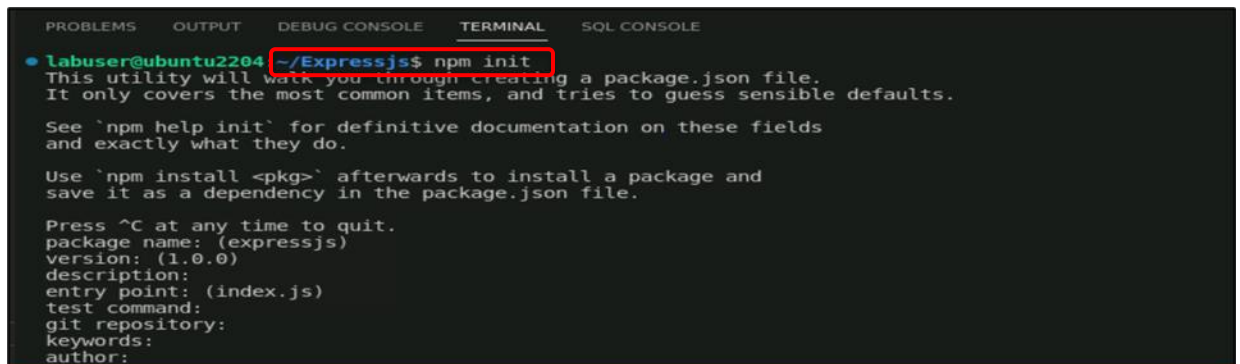
```
mkdir Expressjs  
cd Expressjs
```



A terminal window titled 'Tilix: labuser@ubuntu2204: ~/Expressjs'. The prompt is '1: labuser@ubuntu2204: ~/Expressjs'. The user enters 'mkdir Expressjs' and 'cd Expressjs', both of which are highlighted with red boxes. The prompt then changes to 'labuser@ubuntu2204: ~/Expressjs\$'.

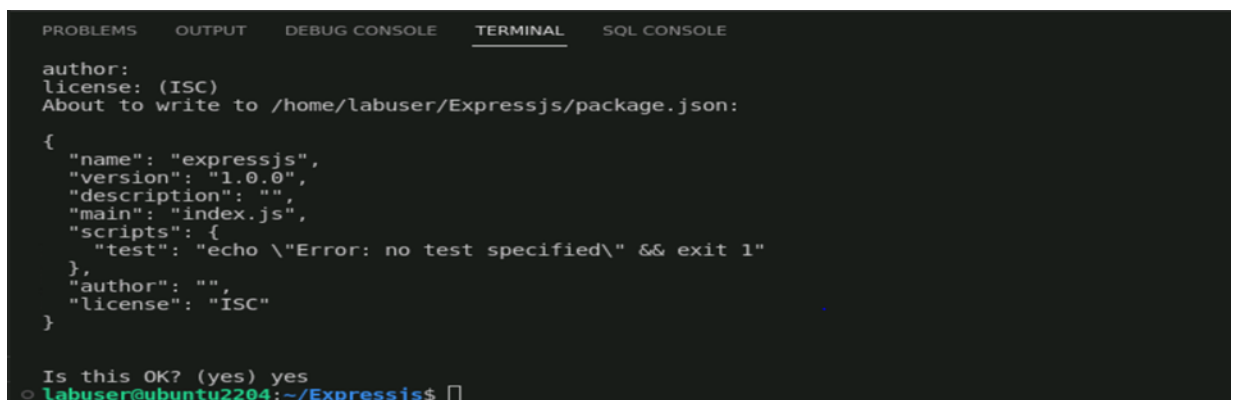
- 2.2 Open the working directory in VS code, open the terminal, and run the below command to create a **package.json** file for the application:

```
npm init
```



A terminal window showing the output of 'npm init'. The prompt is 'labuser@ubuntu2204: ~/Expressjs\$ npm init'. The output includes instructions on how to use 'npm init' and 'npm install', followed by a series of questions with default values: 'package name: (expressjs)', 'version: (1.0.0)', 'description:', 'entry point: (index.js)', 'test command:', 'git repository:', 'keywords:', and 'author:'. The 'npm init' command is highlighted with a red box.

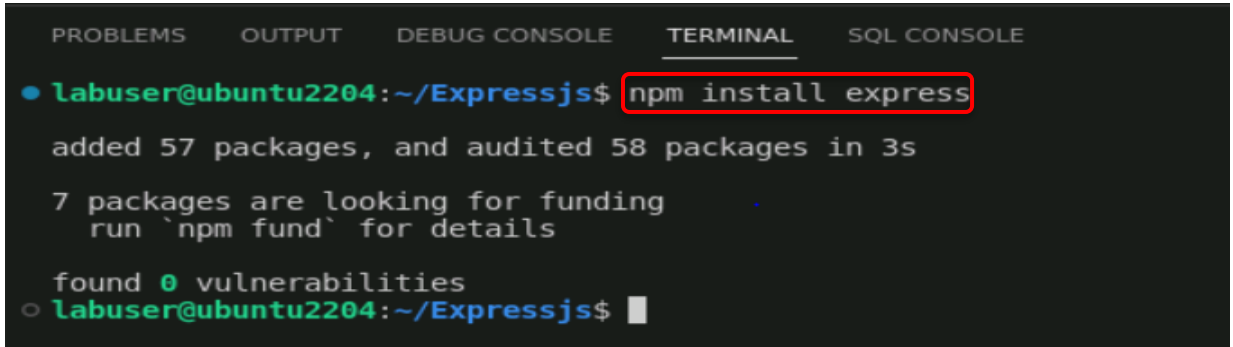
**Note:** Enter all the questions and keep default values for now



A terminal window showing the final output of 'npm init'. The prompt is 'labuser@ubuntu2204: ~/Expressjs\$'. The output shows the default values for the package.json file: 'author:', 'license: (ISC)', and 'About to write to /home/labuser/Expressjs/package.json:'. The output is a JSON object: '{ "name": "expressjs", "version": "1.0.0", "description": "", "main": "index.js", "scripts": { "test": "echo \\\"Error: no test specified\\\" && exit 1" }, "author": "", "license": "ISC" }'. The prompt then asks 'Is this OK? (yes) yes' and the user enters 'yes'. The prompt then changes to 'labuser@ubuntu2204: ~/Expressjs\$'.

2.3 Run the following code to install Express in the Expressjs directory and save it in the dependencies list

**npm install express**



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL SQL CONSOLE
labuser@ubuntu2204:~/Expressjs$ npm install express
added 57 packages, and audited 58 packages in 3s
7 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
labuser@ubuntu2204:~/Expressjs$
```

### Step 3: Create a Hello World Express app

3.1 Create file **index.js** in the Expressjs directory and write the following code for **Hello World** in that file:

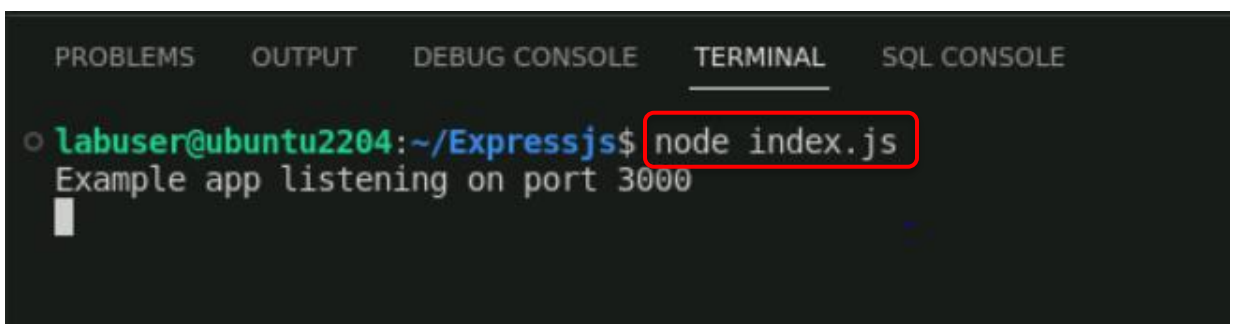
```
const express = require('express')
const app = express()
const port = 3000

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

app.listen(port, () => {
  console.log(`Example app listening on port ${port}`)
})
```

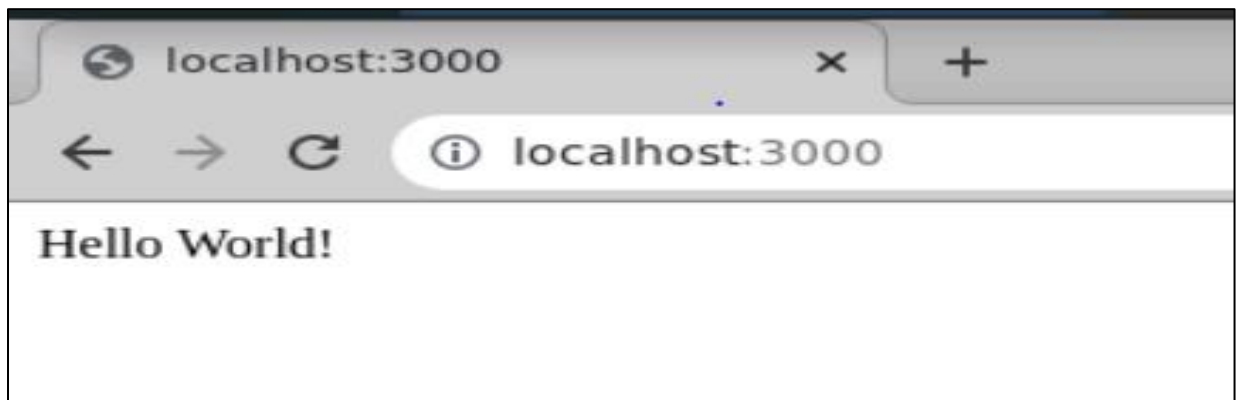
3.2 Run the listed below command in the **terminal** to run the file:

**node index.js**



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL SQL CONSOLE
labuser@ubuntu2204:~/Expressjs$ node index.js
Example app listening on port 3000
```

3.3 Open the browser and search <http://localhost:3000/>.



The above output shows that the **Hello World Express app** is running successfully.

By following these steps, you have successfully installed Express.js, verified Node.js installation, and created a Hello World Express app, demonstrating basic functionality and server setup.