

Introduction to the eCommerce Application Practice Problems Solutions

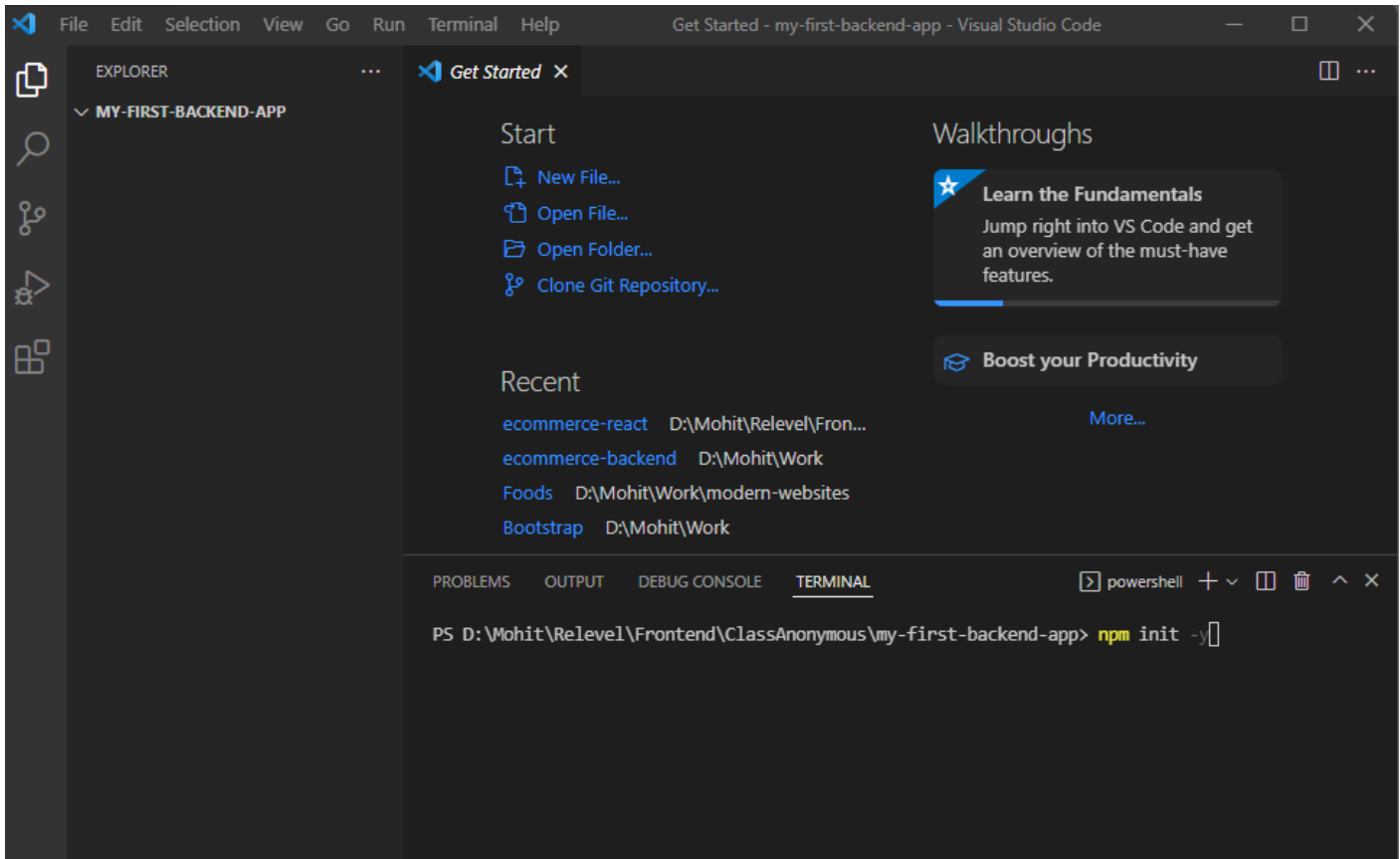


Try creating a new application by initializing a npm project directory and install express in it.

Follow the below steps

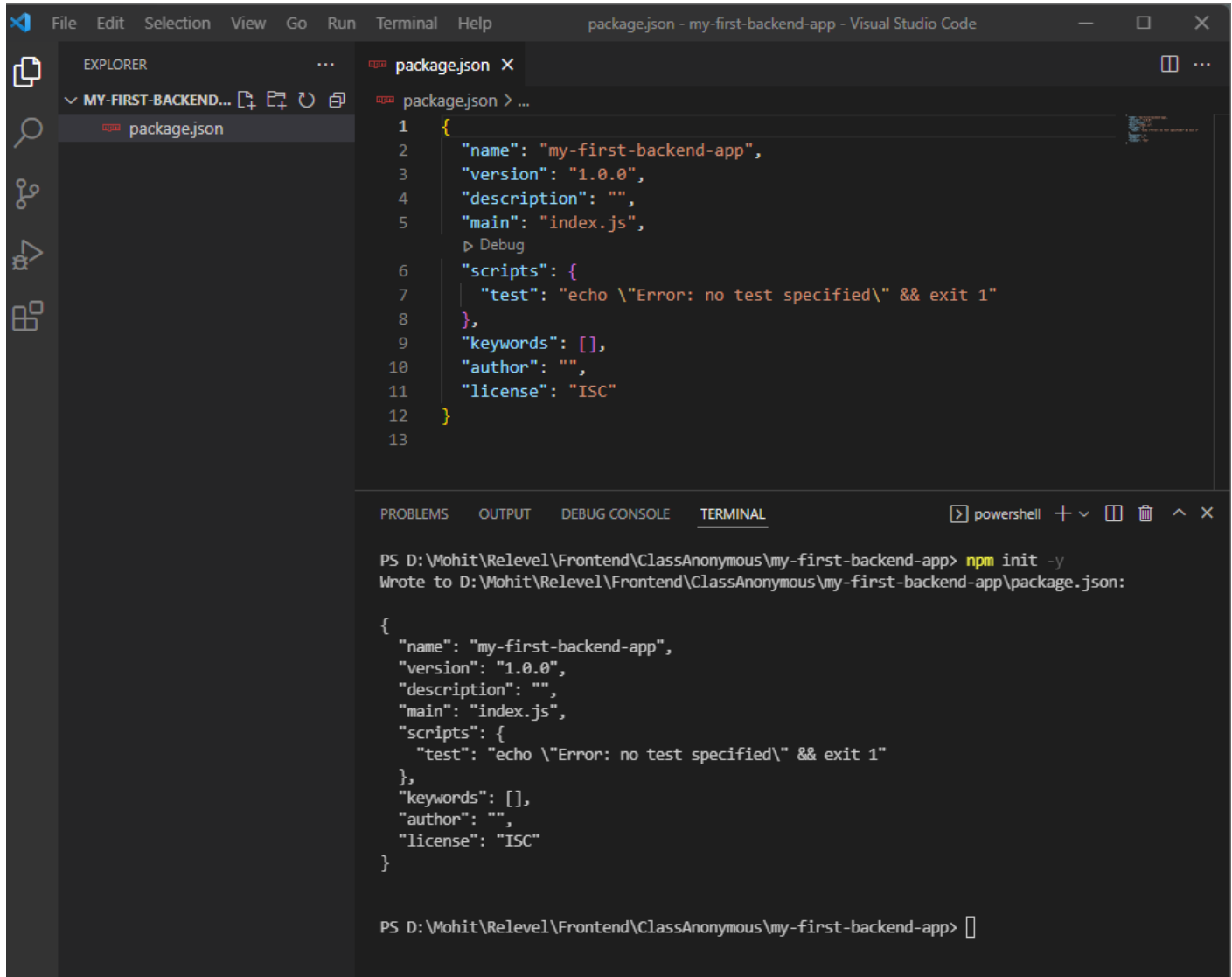
Step 1: Create a directory with your desired name or 'my-first-backend-app'

Step 2: Open the folder using vscode application and press Ctrl + ` OR open command prompt from inside the folder.



Step 3: Now run command “npm init -y” from and answer
This will generate an empty npm project without going through all sorts of questions.

The -y here stands for yes



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows a file named `package.json` under the project `MY-FIRST-BACKEND...`. The main editor displays the content of `package.json`, which is a JSON object with the following properties:

```
1 {
2   "name": "my-first-backend-app",
3   "version": "1.0.0",
4   "description": "",
5   "main": "index.js",
6   "scripts": {
7     "test": "echo \"Error: no test specified\" && exit 1"
8   },
9   "keywords": [],
10  "author": "",
11  "license": "ISC"
12 }
13
```

The Terminal panel at the bottom shows the command `npm init -y` being executed in a PowerShell window. The output indicates that the command wrote the `package.json` file to the project directory, followed by the JSON content shown in the editor.

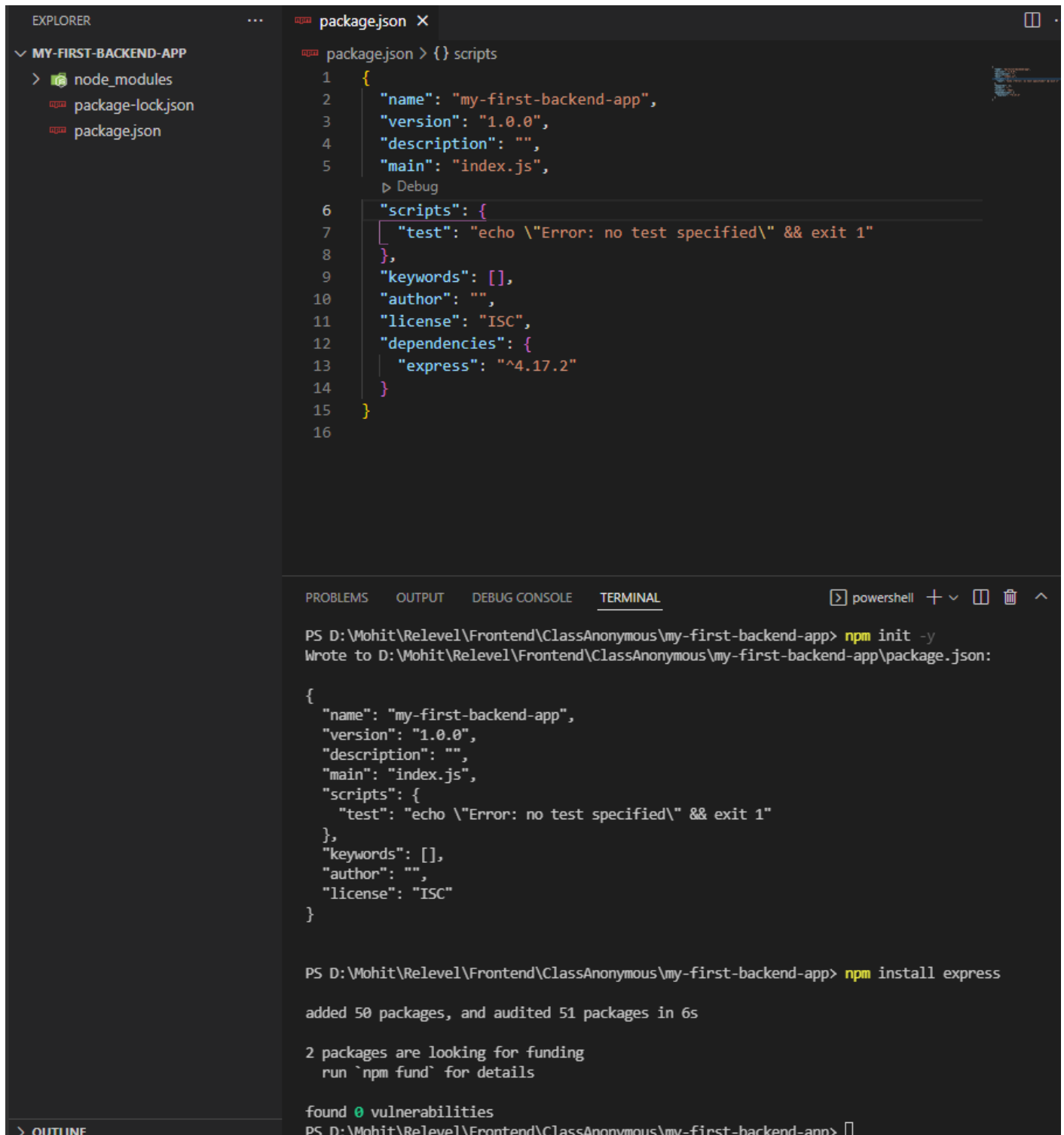
```
PS D:\Mohit\Relevel\Frontend\ClassAnonymous\my-first-backend-app> npm init -y
Wrote to D:\Mohit\Relevel\Frontend\ClassAnonymous\my-first-backend-app\package.json:

{
  "name": "my-first-backend-app",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}

PS D:\Mohit\Relevel\Frontend\ClassAnonymous\my-first-backend-app>
```

Our project is now initialized with a `package.json` file and some default configurations.

Step 4: Now run npm install express to install express on our app



The screenshot shows a VS Code editor with a project named 'MY-FIRST-BACKEND-APP'. The Explorer sidebar on the left shows the file structure with 'node_modules', 'package-lock.json', and 'package.json'. The main editor displays the 'package.json' file with the following content:

```
1  {
2    "name": "my-first-backend-app",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1"
8    },
9    "keywords": [],
10   "author": "",
11   "license": "ISC",
12   "dependencies": {
13     "express": "^4.17.2"
14   }
15 }
```

The bottom panel shows the 'TERMINAL' output with the following commands and results:

```
PS D:\Mohit\Relevel\Frontend\ClassAnonymous\my-first-backend-app> npm init -y
Wrote to D:\Mohit\Relevel\Frontend\ClassAnonymous\my-first-backend-app\package.json:

{
  "name": "my-first-backend-app",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}

PS D:\Mohit\Relevel\Frontend\ClassAnonymous\my-first-backend-app> npm install express

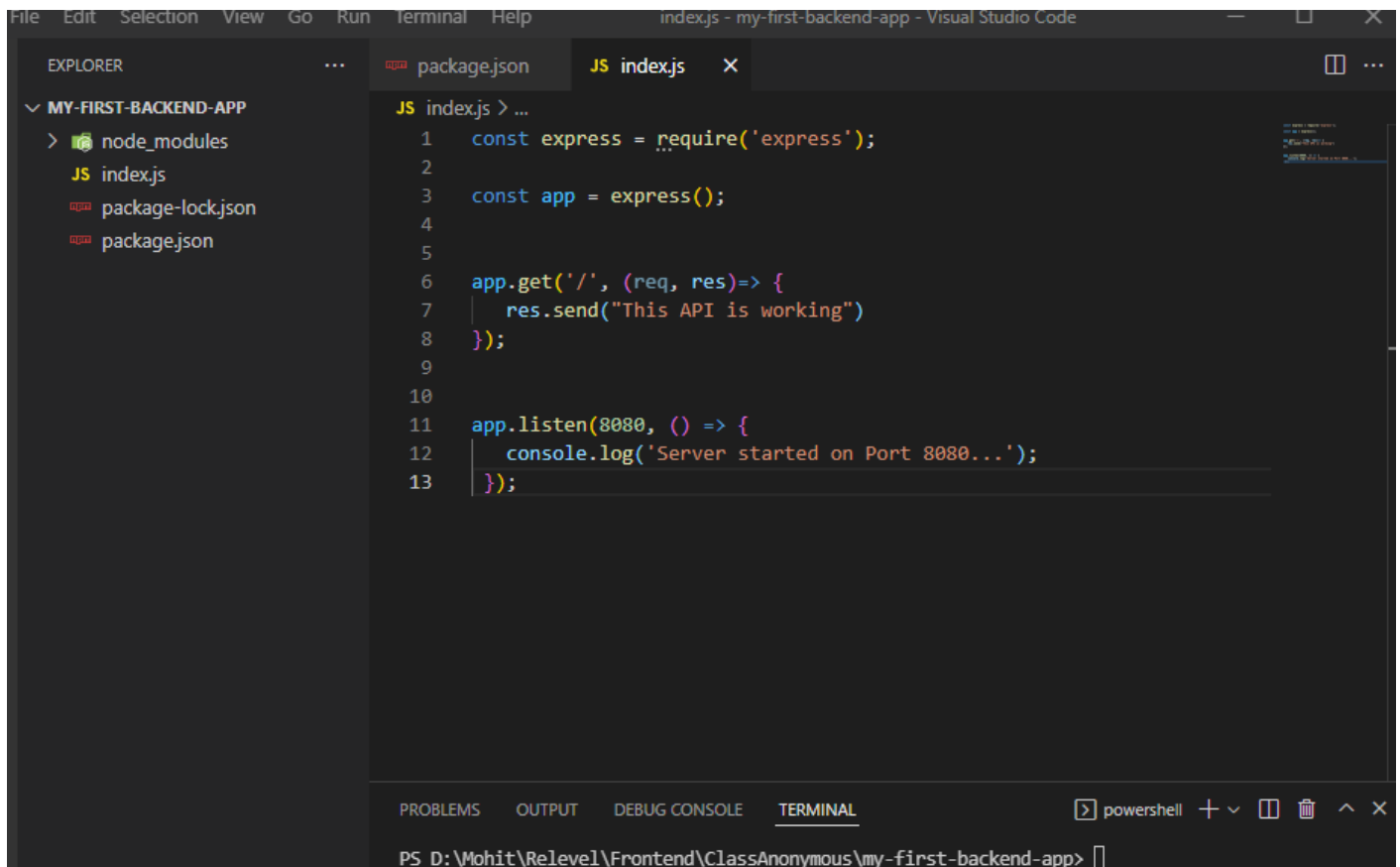
added 50 packages, and audited 51 packages in 6s

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

found 0 vulnerabilities
PS D:\Mohit\Relevel\Frontend\ClassAnonymous\my-first-backend-app>
```

Now, write using express, make the application listen on port 8080 and send a message as “This API is working” from it.

Step 1: Create index.js file and add the below script in it.



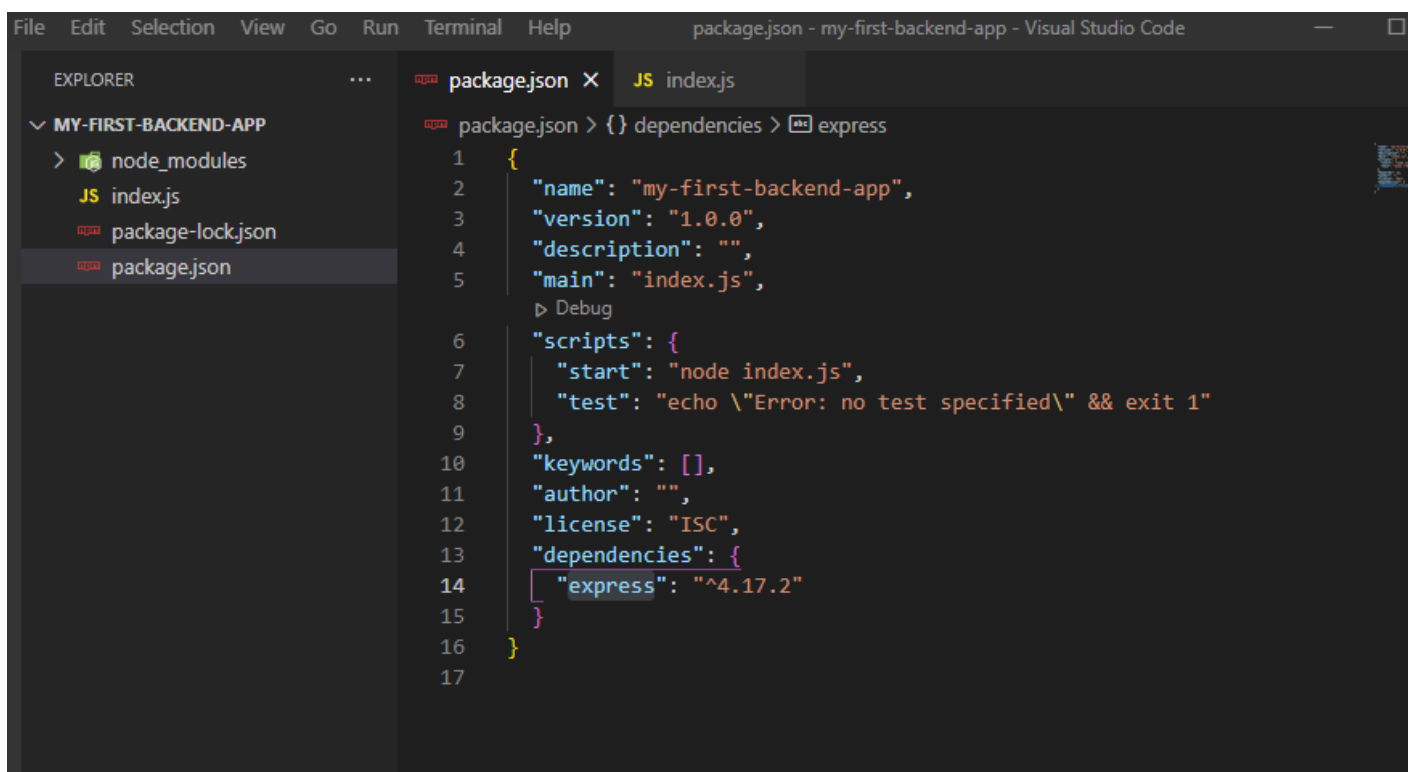
```
File Edit Selection View Go Run Terminal Help index.js - my-first-backend-app - Visual Studio Code

EXPLORER
MY-FIRST-BACKEND-APP
  node_modules
  JS index.js
  package-lock.json
  package.json

JS index.js > ...
1  const express = require('express');
2
3  const app = express();
4
5
6  app.get('/', (req, res) => {
7    res.send("This API is working")
8  });
9
10
11 app.listen(8080, () => {
12   console.log('Server started on Port 8080...');
13 });

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS D:\Mohit\Relevel\Frontend\ClassAnonymous\my-first-backend-app>
```

Step 2: Add start script in package.json file

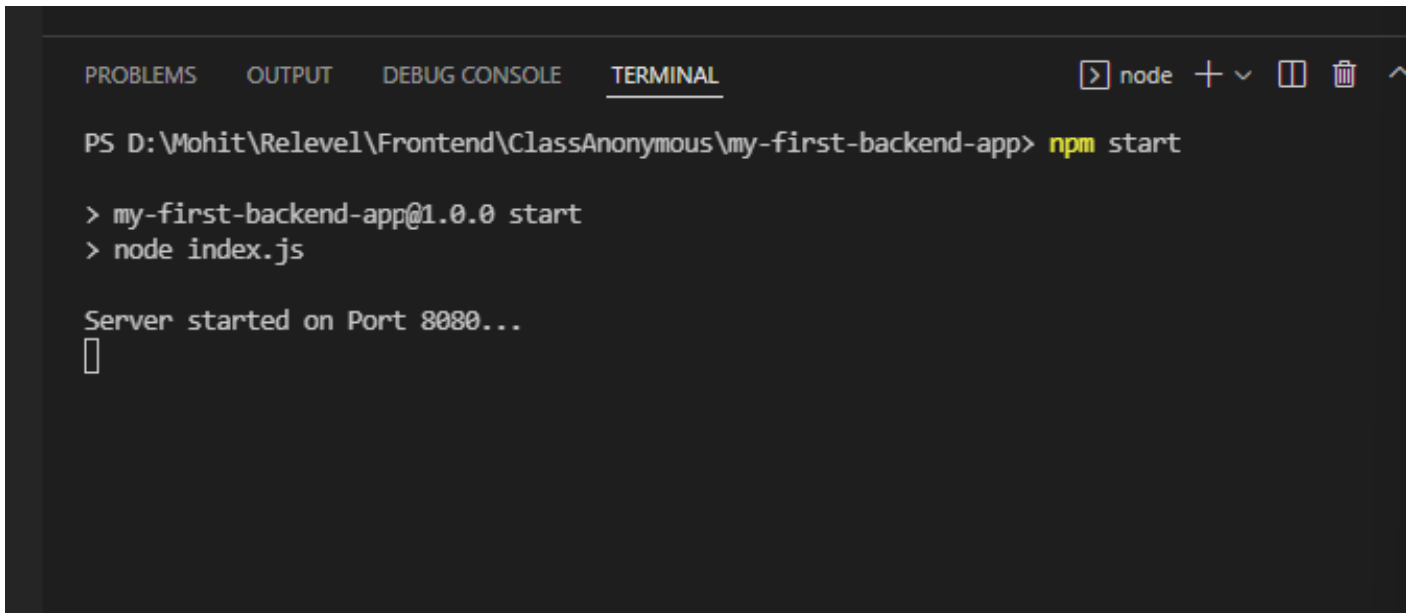


```
File Edit Selection View Go Run Terminal Help package.json - my-first-backend-app - Visual Studio Code

EXPLORER
MY-FIRST-BACKEND-APP
  node_modules
  JS index.js
  package-lock.json
  package.json

package.json > {} dependencies > express
1  {
2    "name": "my-first-backend-app",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "start": "node index.js",
8      "test": "echo \\\"Error: no test specified\\\" && exit 1"
9    },
10   "keywords": [],
11   "author": "",
12   "license": "ISC",
13   "dependencies": {
14     "express": "^4.17.2"
15   }
16 }
17
```

Step 3: run “npm start” command from terminal



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  node + - [] [] ^

PS D:\Mohit\Relevel\Frontend\ClassAnonymous\my-first-backend-app> npm start

> my-first-backend-app@1.0.0 start
> node index.js

Server started on Port 8080...
█
```

Step 4: Now try accessing localhost:8080 port which should look like this

