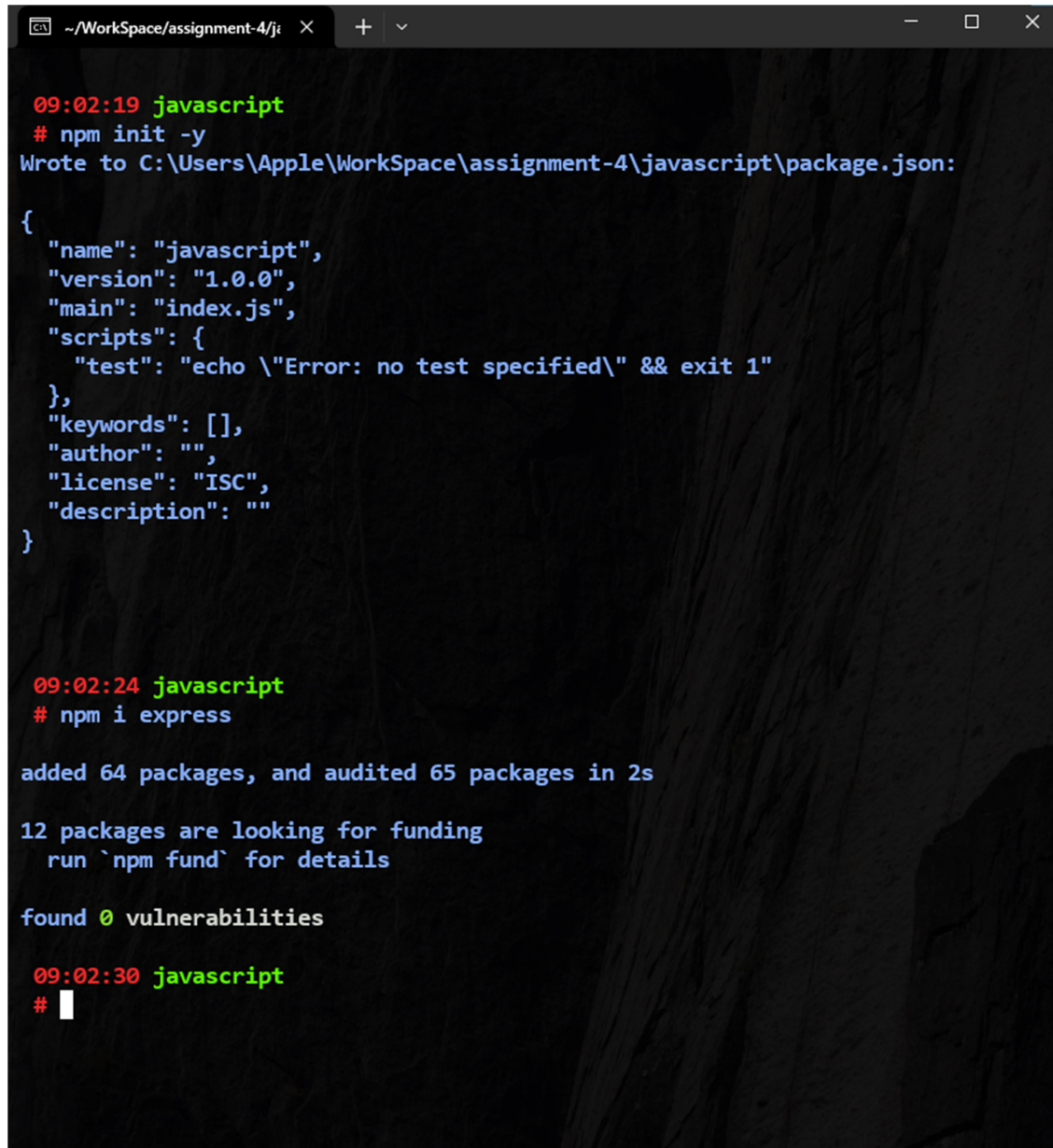


## A Simple http-server using node and express

### Setp1:

Initialize an empty node project using “npm init”, and install express library. Using command “npm install express”.

A terminal window with a dark background and light-colored text. The window title bar shows the file path ~/Workspace/assignment-4/ji and standard window controls. The terminal output shows the execution of 'npm init -y' which creates a package.json file. The content of package.json is displayed in a formatted JSON structure. Following this, 'npm i express' is executed, and the terminal shows the installation progress, including the number of packages added and audited, and a message about funding. The session ends with the prompt '#' and a cursor.

```
09:02:19 javascript
# npm init -y
Wrote to C:\Users\Apple\Workspace\assignment-4\javascript\package.json:

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

09:02:24 javascript
# npm i express

added 64 packages, and audited 65 packages in 2s

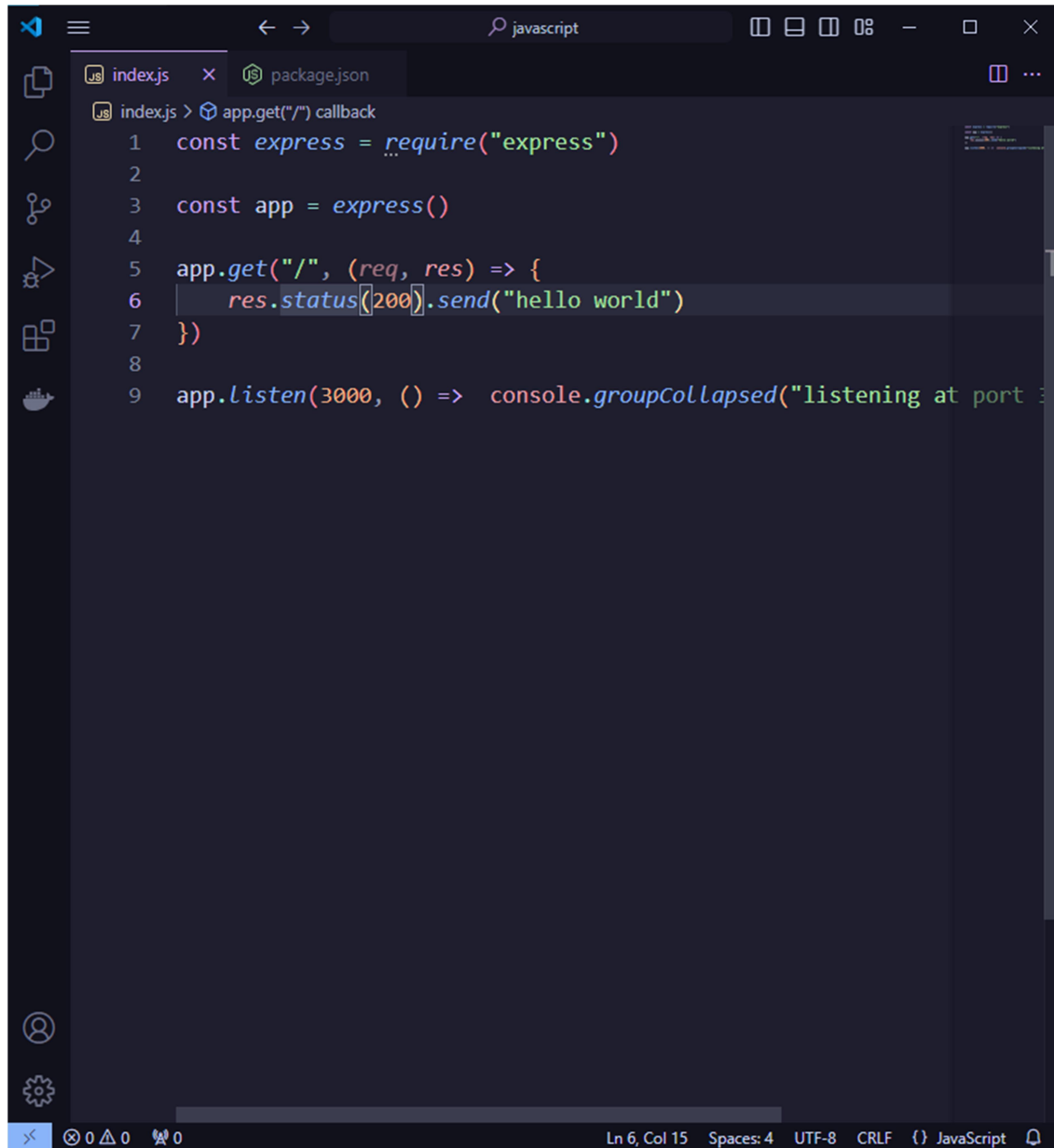
12 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

09:02:30 javascript
#
```

Step2:

Write a simple http-server using express.



The image shows a screenshot of the Visual Studio Code editor with a dark theme. The editor has two tabs open: 'index.js' and 'package.json'. The 'index.js' file is active and contains the following JavaScript code:

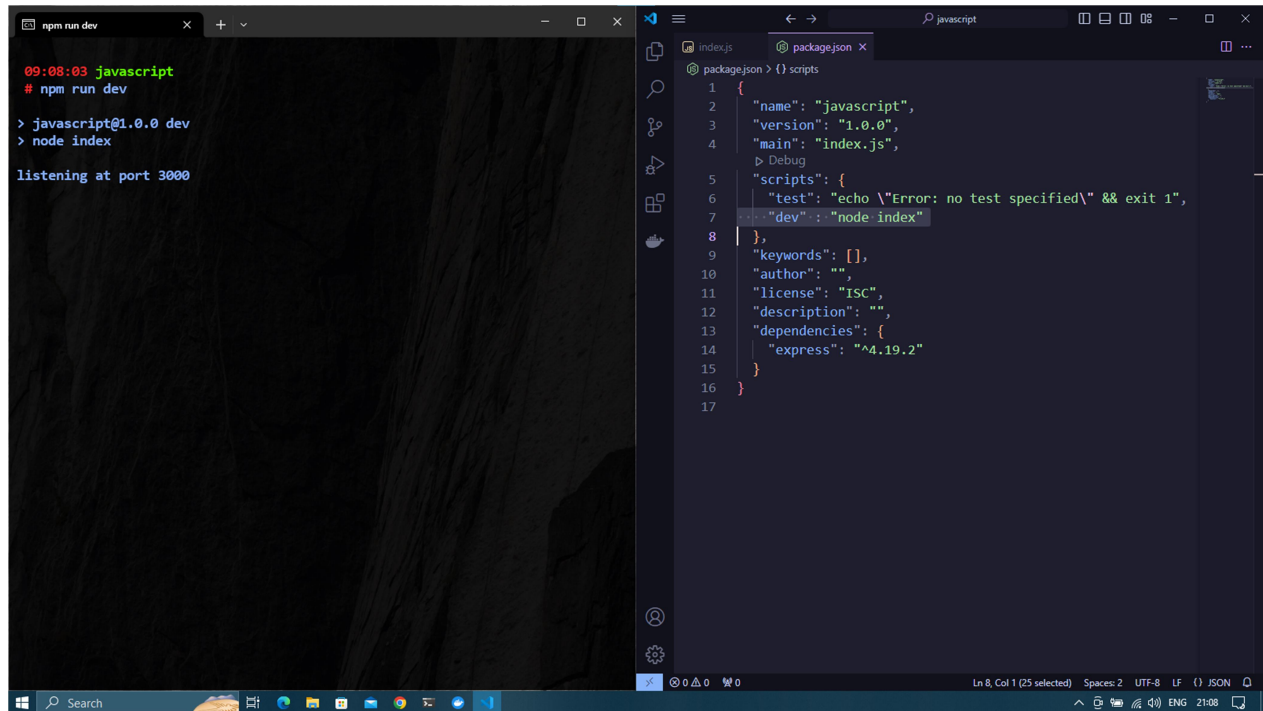
```
1  const express = require("express")
2
3  const app = express()
4
5  app.get("/", (req, res) => {
6    res.status(200).send("hello world")
7  })
8
9  app.listen(3000, () => console.groupCollapsed("listening at port 3000"))
```

The code is written in a standard JavaScript style with 4-space indentation. The status code 200 is highlighted with a blue selection box. The status bar at the bottom indicates the current position is Line 6, Column 15, with 4 spaces, using UTF-8 encoding and CRLF line endings for a JavaScript file.

Step4:

Make changes in package.json, add “dev : node index.js” in scripts object.

After that start the server using command “npm run dev”



The screenshot shows a VS Code editor with two panels. The left panel is a terminal window titled 'npm run dev' showing the following output:

```
09:08:03 javascript
# npm run dev

> javascript@1.0.0 dev
> node index

listening at port 3000
```

The right panel shows the 'package.json' file with the following content:

```
1 {
2   "name": "javascript",
3   "version": "1.0.0",
4   "main": "index.js",
5   "scripts": {
6     "test": "echo \\\"Error: no test specified\\\" && exit 1",
7     "dev": "node index"
8   },
9   "keywords": [],
10  "author": "",
11  "license": "ISC",
12  "description": "",
13  "dependencies": {
14    "express": "^4.19.2"
15  }
16 }
17
```

The status bar at the bottom indicates 'Ln 8, Col 1 (25 selected)', 'Spaces: 2', 'UTF-8', 'LF', and 'JSON'.