

# Practical 9

Link to my GitHub repository:

<https://github.com/RaviThakur322/Node-JS-Practical-GF2023485999/tree/main>

Goal: Create a tool that scans node\_modules, computes SHA-256 of installed package tarballs (or files), and builds a dependency graph with license detection.

Task: Implement a script that:

Reads top-level node\_modules/\*/package.json

Recursively resolves dependencies to build a graph

Reports packages without license file

## 1. scan\_node\_modules.js

```
JS moduleA.mjs U JS moduleB.js U {} data.json U Practical 7.pdf U JS sandbox-installer.js U Readme.txt Practical 9 U JS scan_node_modules.js U X
Practical 9 > JS scan_node_modules.js > ...
1 import { promises as fs } from 'fs';
2 import path from 'path';
3
4 // --- Configuration ---
5
6 // List of common license file names. Case-insensitive.
7 const LICENSE_FILES = [
8   'LICENSE',
9   'LICENSE.MD',
10  'LICENSE.TXT',
11  'LICENSE.RST',
12  'UNLICENSE',
13  'UNLICENSE.MD',
14  'COPYING',
15  'COPYING.MD',
16 ];
17
18 // --- Main Scanner Logic ---
19
20 /**
21  * Finds a license file in a given package directory.
22  * @param {string} packagePath - The absolute path to the package directory.
23  * @returns {Promise<boolean>} - True if a license file is found, false otherwise.
24  */
25 async function findLicenseFile(packagePath) {
26   try {
27     const entries = await fs.readdir(packagePath, { withFileTypes: true });
28     for (const entry of entries) {
29       if (entry.isFile()) {
30         const entryNameUpper = entry.name.toUpperCase();
31         if (LICENSE_FILES.includes(entryNameUpper)) {
32           return true;
33         }
34       }
35     }
36   } catch (err) {
37     console.error(`[Error] Could not read directory: ${packagePath}`, err.message);
38   }
39   return false;
40 }
```

## Output:

```
--- Scan Complete ---  
Total unique packages found: 68
```

```
[Success] All packages have a license file!
```

```
>
```