

# Practical – 9

GitHub Link: <https://github.com/Devsharma511/NodeJs.git>

1. 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.

2. Tasks: Implement a script that:

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

Recursively resolves dependencies to build a graph

Reports packages without license file

Project Structure:

scan\_packages.js:

```
JS scan_packages.js U X
Practical-9 > JS scan_packages.js > ...
1  import fs from 'fs';
2  import path from 'path';
3  import crypto from 'crypto';
4
5  const nodeModulesDir = path.join(process.cwd(), 'node_modules');
6
7  function sha256File(filePath) {
8    const data = fs.readFileSync(filePath);
9    return crypto.createHash('sha256').update(data).digest('hex');
10 }
11
12 function getPackageInfo(pkgName) {
13   const pkgDir = path.join(nodeModulesDir, pkgName);
14   const pkgJsonPath = path.join(pkgDir, 'package.json');
15   if (!fs.existsSync(pkgJsonPath)) return null;
16   const pkgJson = JSON.parse(fs.readFileSync(pkgJsonPath, 'utf8'));
17   const files = fs.readdirSync(pkgDir);
18   const hasLicense = files.some(f => f.toLowerCase().startsWith('license'));
19   const jsFiles = files.filter(f => f.endsWith('.js')).slice(0, 2);
20   const shaPreview = jsFiles.map(f =>
21     `${f}:${sha256File(path.join(pkgDir, f)).slice(0, 8)}...`
22   ).join(' ');
23   return {
24     version: pkgJson.version,
25     hasLicense,
26     shaPreview,
27     fileCount: files.length
28   };
29 }
30
31 const packages = fs.readdirSync(nodeModulesDir).filter(p =>
32   fs.existsSync(path.join(nodeModulesDir, p, 'package.json'))
33 );
34
35 console.log('Dependency Summary:');
36 packages.forEach(pkg => {
37   const info = getPackageInfo(pkg);
38   if (!info) return;
39   console.log(`- ${pkg} (version: ${info.version}, files: ${info.fileCount}, license: ${info.hasLicense ? 'Yes' : 'No'}, sha: ${info.shaPreview})`);
40 });
41
42 console.log('\nPackages missing license file:');
43 packages.forEach(pkg => {
44   const info = getPackageInfo(pkg);
45   if (info && !info.hasLicense) {
46     console.log(pkg);
47   }
48 });
49
```

package-lock.json:

```

1  {
2    "name": "Practical-9",
3    "lockfileVersion": 3,
4    "requires": true,
5    "packages": {
6      "": {
7        "dependencies": {
8          "lodash": "^4.17.21"
9        }
10     },
11     "node_modules/lodash": {
12       "version": "4.17.21",
13       "resolved": "https://registry.npmjs.org/lodash/-/lodash-4.17.21.tgz",
14       "integrity": "sha512-v2kDEe571ecTulaDIuNTPy3Ry4gLGJ6Z103vE1krqXZNrsQ+LFTGHVxVjcXP17LhZVVGedAJv8X21tvj5FvSg==",
15       "license": "MIT"
16     }
17   }
18 }
19

```

Output:

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

powershell - Practical-9 + ▢ ▢ ... | [X] X

Dependency Summary:  
- lodash (version: 4.17.21, files: 640, license: Yes, sha: add.js:62192fb4..., after.js:ca045b1a...)

Packages missing license file:  
(node:4536) [MODULE\_TYPELESS\_PACKAGE\_JSON] Warning: Module type of file:///C:/Users/sharm/nodejs/Practical-9/scan\_packages.js is not specified and it doesn't parse as CommonJS  
.  
Reparsing as ES module because module syntax was detected. This incurs a performance overhead.  
To eliminate this warning, add "type": "module" to C:\Users\sharm\nodejs\Practical-9\package.json.  
(Use `node --trace-warnings ...` to show where the warning was created)  
PS C:\Users\sharm\nodejs\Practical-9>