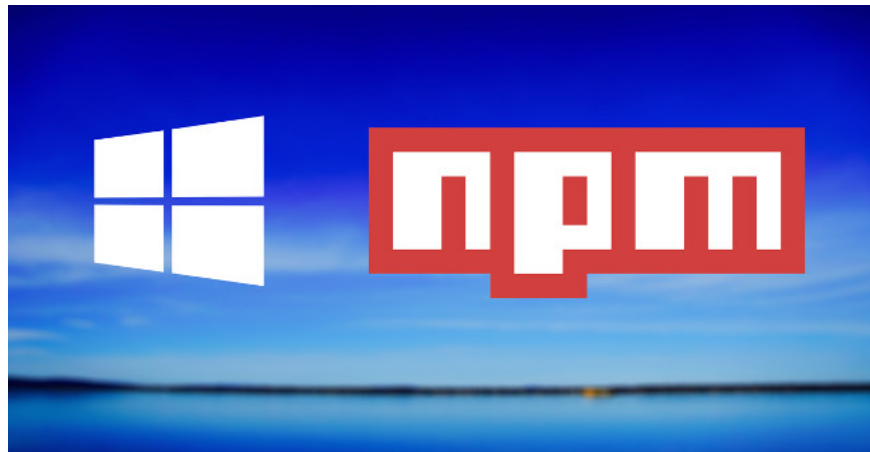


# What is npm shrinkwrap and when should it be used?

Do you know what npm shrinkwrap is and when to use it? Please refer to this article!

1. What is Java? Why choose Java?
2. What is WebAssembly?
3. 9 reasons you should equip a little knowledge of HTML and CSS

The **npm shrinkwrap** command allows you to lock all versions of packages and *descendants of children and grandchildren* in the **node\_modules** directory. This article will explain **why and when to use the npm shrinkwrap command during your application development**.



The package manager npm works quite well in maintaining and installing dependencies for all packages that your project requires. It does that by installing a package hierarchy in the `node_modules` directory.

**There are two main problems with how to set up npm activity:**

1. Although npm encourages the use of semver to **manage the application version** of packages, it depends entirely on the package author's compliance with this principle. This may be a problem if a package in your application does not follow semver, a new version of this package may cause an error.

Even, even if the author of the package follows semver, the **error can still occur in a introduced version that is compatible** .

2. Another problem arises because of **how to set up npm activity** . When running an npm setup install a package hierarchy, if you want to correctly install the package version number, you can use the exact version

number of the packages in the package.json file. However, that only solves the problem for direct dependencies without helping you **control the version of the package packages in the package** and more.

This can be very important to you in a production environment because you need to make sure that every time you deploy the product, you always install the same version of the package as other implementations.

**This is the time you need to use npm shrinkwrap** . When you run the npm shrinkwrap command in a project after running the npm install command, it creates a file called **npm-shrinkwrap.json** , which lists exactly the version of all packages installed in whole package hierarchy. If you move to a version management system when your colleague clone the source code and run the npm install command, it will correctly install the package version in the entire hierarchy, as specified. in the file **npm-shrinkwrap.json** .

To update the npm-shrinkwrap.json file, you will need to run npm update, then determine the exact package you need to run the npm shrinkwrap command to update the file npm-shrinkwrap.json.

If you need to find outdated (outdated) packages, just run:

```
npm outdated
```

The above command will notify you of obsolete packages. You can check them and decide whether or not to include them in the product after a thorough inspection.

Also, note that npm shrinkwrap does not include devDependencies unless you run with the -dev flag:

```
npm shrinkwrap --dev
```

**Refer to some more articles:**

1. Form a way of thinking like a programmer
2. 11 basic principles that every programmer should follow
3. Journey from unknown to becoming software engineer in San Francisco for 12 months

Having fun!

You finished reading the article "**What is npm shrinkwrap and when should it be used?**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.