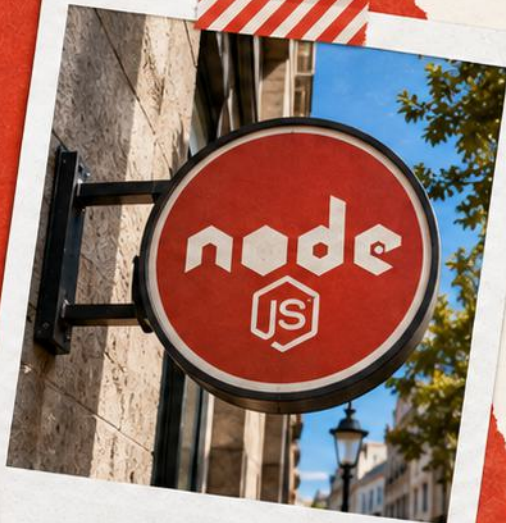




express	4.18.2	1.2M
dotenv	16.4.5	34K
axios	1.6.7	421K
lodash	4.17.21	531K
tailwindcss	3.4.1	3.8M
vue	3.4.15	2.1M
eslint	8.57.0	1.1M
prettier	3.2.5	631K
vite	5.1.6	1.5M

added 256 packages in 12s



node_modules/

Node.js et npm

Node.js fait tourner du JavaScript en dehors du navigateur, et npm installe les outils dont tes projets ont besoin.



Le même JavaScript, exécuté par ton ordinateur.

sans navigateur

Le JavaScript que tu connais s'exécute dans le navigateur. Node.js, c'est le même langage, sans navigateur.

Les outils de dev modernes sont eux-mêmes écrits en JavaScript : Vite, Vue, Tailwind, ESLint ont tous besoin de Node.

Tu ne vas quasiment jamais écrire de code « pour Node ». Tu l'installes une fois.

Le magasin d'applications des outils de dev.

npm = Node Package Manager, le gestionnaire de paquets de Node.

Un paquet, c'est du code écrit par quelqu'un d'autre que tu ajoutes à ton projet.

npm s'installe automatiquement avec Node.js.
Rien à faire de plus.



```
npm install tailwindcss  
npm install vue
```



Deux commandes avant tout le reste.

```
~ % node --version  
v24.21.0
```

```
~ % npm --version  
11.6.2
```

Deux numéros → c'est installé.
Sinon : Windows télécharge le
Standalone Binary (.zip) LTS et
le place dans C:\laragon\bin\nodejs\ ;
Mac télécharge le .pkg LTS, puis
ferme et rouvre le terminal.



Cinq commandes pour toute l'année.

Installer les dépendances • ajouter un paquet • lancer le serveur de dev sur `http://localhost:5173` • préparer la mise en ligne dans `dist/` • démarrer un projet.



On partage la **recette**, pas les courses.

`package.json` est la carte d'identité
du projet : tu le lis, `npm` l'écrit.

`node_modules/` est reconstructible :
jamais sur Git, jamais dans un zip.