



# Node.js

- Node.js est un environnement d'exécution JavaScript
- De même que Python, il vous permet d'exécuter du JavaScript sur votre machine et d'en faire des applications
- Node.js permet, entre autres, de faire
  - Des serveurs web
  - Des applications web
  - Des applications desktop et en ligne de commande
  - Des scripts
  - etc.
- De même que Python possède son gestionnaire de paquets *pip*, Node.js possède *npm* (*node packet manager*)

Node.js est donc une application qui interprète du JavaScript et le transforme en instructions pour votre machine.

Il existe certaines variations entre le JavaScript de Node et celui de votre navigateur. Par exemple Node ne possède pas de DOM, car le script ne s'exécute pas sur le page web. Node vient également avec un ensemble de fonctions permettant d'interagir avec votre système (créer des fichiers, gérer des processus, etc), ce qui ne fait pas de sens pour une page web.

Il existe d'autres environnements d'exécution et d'autres gestionnaires de paquets que Node et npm. Nous utilisons ceux-ci car ce sont les plus répandus et les plus standards. Les alternatives ont souvent des fonctions très similaires.

# Node.js

Quelques commandes de bases

Dans le répertoire d'un projet Node.js :

- `npm init` : Crée un nouveau projet Node.js partant de zéro
- `npm install monPackage` : Installe le packet `monPackage`, peut être abrégé `npm i monPackage`
- `npm install` : Installe toutes les dépendances (packet) du projet, peut être abrégé `npm i`
- `npm run maCommande` : Exécute la commande `maCommande`
- `npm -h` : Affiche l'aide des commandes

Contrairement à Python, par défaut, tous les packets sont installés dans le repertoire du projet.  
Chaque projet contient donc indépendamment tous ses packets, ils ne sont pas partagés entre projets.

# Node.js

## Structure d'un projet

Un projet avec Node.js est notamment composé des fichiers / dossiers :

- **package.json** : Liste les informations principales concernant un projet

```
{
  "name": "monProjet",    // Nom du projet
  "type": "module",      // Son type, généralement un module
  "scripts": {           // Liste des commandes (actions) disponibles pour le projet
    "start": "vite",      // Executer "npm run start" va executer la commande "vite"
    "build": "vite build"
  },
  "dependencies": {       // Liste des dépendances pour les projet, ajoutées au moment d'un "npm install librairie1"
    "librairie1": "^1.0.0",
    "librairie2": "^1.2.0",
    "librairie3": "^2.3.0"
  }
}
```

# Node.js

## Structure d'un projet

Un projet avec Node.js est notamment composé des fichiers / dossiers :

- **package-lock.json** : Liste les sous-dépendances (dépendances de dépendances).  
Est recréé lors d'un *npm install*
- **node\_modules** : Dossier contenant toutes les dépendances téléchargées  
Il ne doit pas être ajouté à votre projet git, car il est souvent très lourd et recréé lors d'un `npm install`

Pour installer et lancer une application qui utilise Node.js, suivez les instructions qui sont généralement du style :

```
git clone <repository> // Clone le projet avec git
npm install             // Installe toutes les dépendances
npm run start           // Lance le script de démarrage (défini dans package.json)
```



# Vite

- Vite est un *bundler* fonctionnant avec Node.js (entre autres)
- Un bundler prend un ensemble de fichiers JavaScript, HTML, CSS, images, etc et produit une application web rapide et minimale
- Les avantages sont :
  - **Gestion moderne des dépendances** : plus besoin d'avoir des fichiers JS inclus dans tous les sens au milieu du HTML. Les dépendances sont installées avec npm et incluses avec du JavaScript moderne
  - **Résultat minimal** : Vite simplifie le code en n'incluant que les parties de code vraiment nécessaires et minimise le code
  - **Support d'outils d'automation et d'integration continue** (mise en forme automatique, tests, script lors d'un commit, etc)
  - **Ajout facile d'autres technologies** (Vue, TypeScript, SCSS, Svelte, React, etc)

Ce bundler est tellement utilisé que certaines librairies en font leur moyen de diffusion principal. C'est le cas d'OpenLayers.