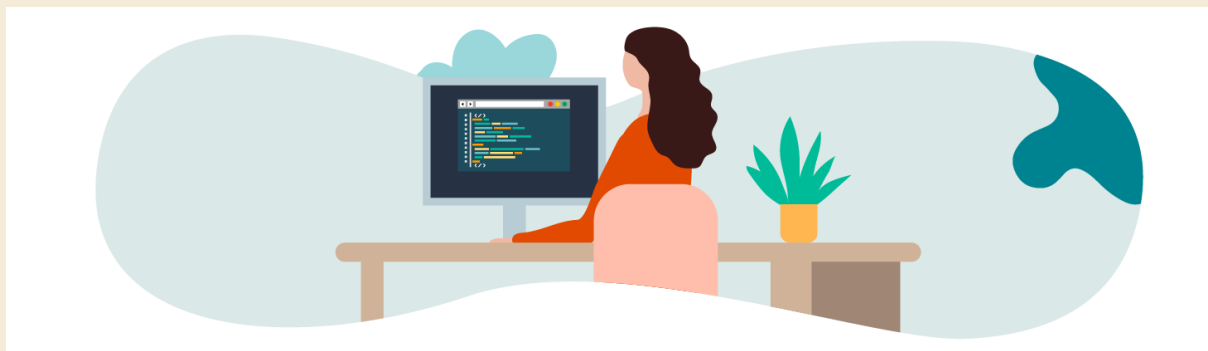


# Installez la librairie grâce au gestionnaire de paquets NPM

5–6 minutes



Votre choix est fait, vous allez utiliser la librairie Chart.js pour faciliter l’affichage de graphiques sur votre page web de gestion. Cependant, les fichiers de la librairie ne sont pas encore installés sur votre ordinateur. Voyons ensemble comment faire, avec le gestionnaire de paquets NPM et la ligne de commande.

## Utilisez un gestionnaire de paquets

La plupart du temps, en JavaScript, les développeurs gèrent le téléchargement des librairies avec un **gestionnaire de paquets**.

Mais... Pourquoi on parle de gestionnaire de “paquets” alors que je veux juste télécharger une librairie ?

Parce que les librairies doivent être **empaquetées** pour être distribuées au monde entier. C’est un peu la même idée que d’emballer votre colis et de le mettre dans un carton avant de l’envoyer par la poste.

## Découvrez le rôle d’un gestionnaire de paquets

Pour notre site web, nous utiliserons le gestionnaire de paquets le

plus répandu dans le monde du développement web en JavaScript.

Il s'agit de **NPM** : Node Package Manager. Ce gestionnaire est composé de deux parties :

- le site web public qui référence tous les paquets publiés ;
- l'outil en ligne de commande sur votre ordinateur.

Le site web NPM est accessible à [cette adresse](#), et l'outil en ligne de commande est installé en même temps que vous installez Node.js sur votre ordinateur.

Un gestionnaire de paquets, et NPM en particulier, a plusieurs fonctions au sein d'un projet :

- documenter le nom et la version des librairies dont votre projet dépend. C'est pour cela qu'on les appelle parfois des *dépendances* ;
- automatiser le téléchargement de ces dépendances auprès du site web public qui référence tous les paquets ;
- orchestrer les tâches de maintenance du projet telles que la construction du site web (build), la réalisation de tests ou la vérification du code.

## Initialisez NPM dans votre projet

Pour documenter les librairies dont nous avons besoin, nous devons initialiser un nouveau projet avec NPM.

Si ce n'est pas déjà fait, créez un répertoire pour votre projet, puis lancez la commande `npm init` :

```
# Création d'un répertoire (optionnel)
mkdir mon-projet
# Changement de répertoire
cd mon-projet
# Initialisation du projet NPM
npm init
```

La commande d'initialisation vous demande de saisir un certain

nombre d'informations, comme le nom de votre projet, la version, une description, l'auteur ou bien la licence logicielle. Elle affiche un aperçu du fichier package.json qui va être généré, et vous demande de confirmer.

Une fois confirmé, vous obtiendrez donc le fichier suivant :

```
{
  "name": "site-web",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "dev": "http-server"
  },
  "author": "Thomas Kerbrat",
  "license": "ISC",
  "dependencies": {
    "http-server": "^14.1.0"
  }
}
```

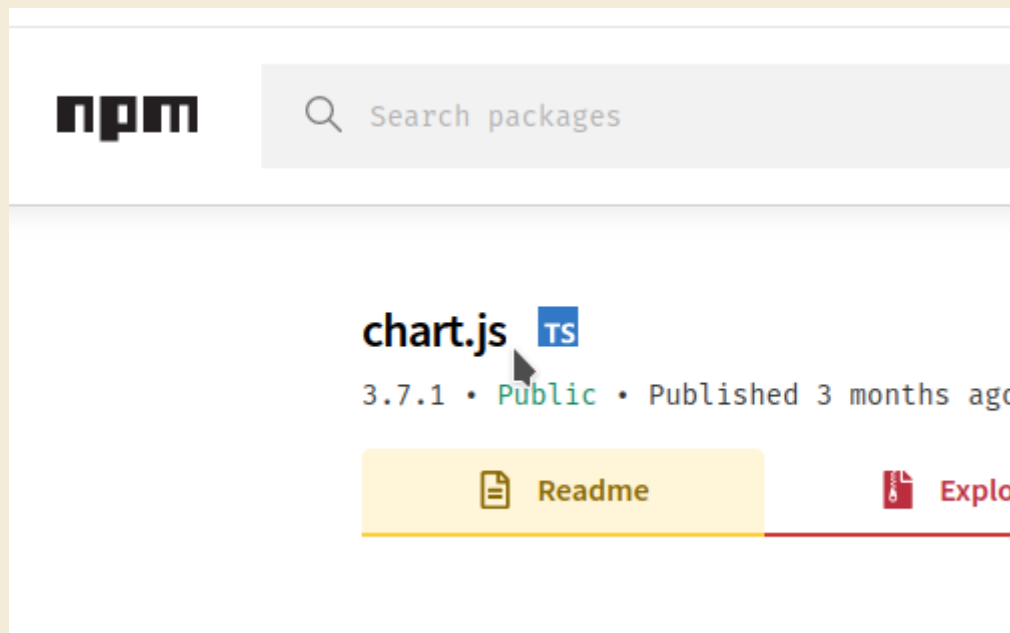
Ce fichier utilise le format JSON. On y retrouve un objet avec des propriétés correspondant aux informations demandées précédemment.

## Installez la librairie

Maintenant que vous avez initialisé votre projet avec NPM, il ne vous reste plus qu'à installer la librairie Chart.js.

Pour y parvenir, nous devons trouver le nom exact du paquet tel que npmjs.com le connaît. On peut facilement trouver ce nom sur [la page de description du paquet Chart.js](#). Il se situe en haut à

gauche, au-dessus des onglets “Readme”, “Explore”, etc. :



Nom officiel et exact de la librairie Chart.js sur NPM

Muni de cette information, placez-vous maintenant dans le répertoire de votre projet, et avec l’interface textuelle, tapez la commande suivante :

```
npm install chart.js
```

NPM vous affiche des informations concernant sa progression de téléchargement et d’installation du paquet demandé. Si tout s’est bien passé, vous devriez obtenir un message similaire à celui-ci :

```
tkerbrat@home-pc:~/site-web$ npm install chart.js
added 40 packages, and audited 41 packages in 498ms

9 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
tkerbrat@home-pc:~/site-web$
```

Résultat de la commande `npm install chart.js`

La commande NPM crée un répertoire `node_modules` (s’il n’est pas déjà présent) dans lequel elle place tous les fichiers de la librairie. Ce répertoire contient un sous-dossier par librairie installée dans votre projet.

En dehors du répertoire `node_modules`, la commande NPM

mémorise que la librairie a été installée en ajoutant son nom dans le fichier `package.json`. Vous devrez versionner ce fichier dans l'historique Git afin de documenter les librairies utilisées dans votre projet.

## Récapitulons en vidéo

Vous pouvez revoir les différentes étapes de cette démonstration dans la vidéo ci-dessous :

## En résumé

- Utiliser un gestionnaire de paquets vous permet de :
- documenter les librairies dont dépend votre projet ;
- automatiser le téléchargement de ces dépendances ;
- orchestrer les tâches routinières.
- Lancez la commande **`npm install nom_du_paquet`** pour :
- enregistrer le paquet comme une dépendance ;
- télécharger les fichiers nécessaires automatiquement.

*Tout est en place pour enfin ajouter un graphique sur notre page web. Retrouvez-moi au chapitre suivant pour apporter la touche finale au site des Bonnes Pièces !*