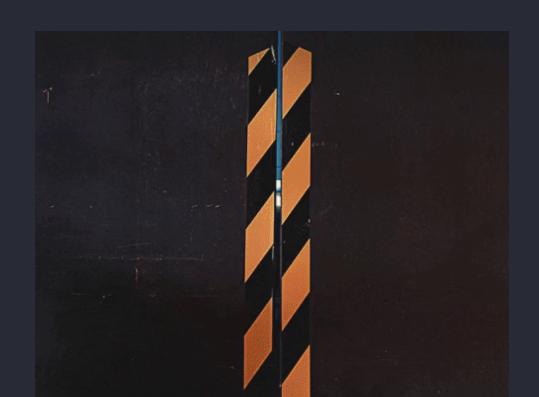
Comment écrire un bon README sur les projets GitHub



README mais pour quoi faire ?

- 1. Un bon projet = un fichier README détaillé.
- 2. Le fichier **README** peut servir de *"guide"* pour l'utilisateur.
- 3. C'est essentile car c'est le premier fichier vu (affiché par GitHub) lorsqu'on arrive sur le projet.
- 4. Cela aidera à nous concentrer sur ce que le projet doit livrer et comment.
- Peut être écrit en plus format mais nous = Markdown

Important

Il n'y a pas qu'une seule bonne façon de structurer un bon README. Mais ...

il y a une très mauvaise façon, et c'est de ne pas inclure de README du tout.

Avant de commencer

Le README doit répondre aux questions Quoi ?, Pourquoi ?, Comment ?



Qu'est ce que doit inclure le README



- 1. Titre du projet
- 2. Description du projet
- 3. (Optionenl) Table des matières
- 4. Comment l'installer et exéctuer le projet ?
- 5. Comment utiliser le projet ?
- 6. (Optionnel) Ajouter une licence
- 7. (Optionnnel) Badges

Exemple (faker)



Faker

Generate massive amounts of fake (but realistic) data for testing and development.



Try it Online

Open in StackBlitz

- API Documentation
- 1 You are reading the docs for the next branch 1

- W
- Locations Generate valid looking Addresses, Zip Codes, Street Names, States, and Countries!
- 💯 Time-based Data Past, present, future, recent, soon... whenever!
- Cocalization Set a locale to generate realistic looking Names, Addresses, and Phone Numbers.
- 👺 Finance Create stubbed out Account Details, Transactions, and Crypto Addresses.
- Names, Adjectives, and Descriptions.
- Hacker Jargon "Try to reboot the SQL bus, maybe it will bypass the virtual application!"
- 🕴 Names Generate virtual humans with a complete online and offline identity.
- 🔢 Numbers Of course, we can also generate random numbers and strings.
- (i) Note: Faker tries to generate realistic data and not obvious fake data. The generated names, addresses, emails, phone numbers, and/or other data might be coincidentally valid information. Please do not send any of your messages/calls to them from your test setup.

Install

```
npm install --save-dev @faker-js/faker
```

Usage

```
// ESM
import { faker } from '@faker-js/faker';

// CJS
const { faker } = require('@faker-js/faker');
```

Markdown

```
# Titre du projet
blablabla description
Pas hésiter à mettre des schémas (rest api ?)
## 🂗 Installation
```bash
npm install
ou
yarn
💢 Usage
```

### Conclusion

• Faire un README c'est bien!



## Références

- (Version française) Comment écrire un bon fichier README pour votre projet GitHub
- How to Write a Good README File for Your GitHub Project
- (GitHub) faker-js