

Jean-Frédéric VINCENT

GIT & GitHub



Git

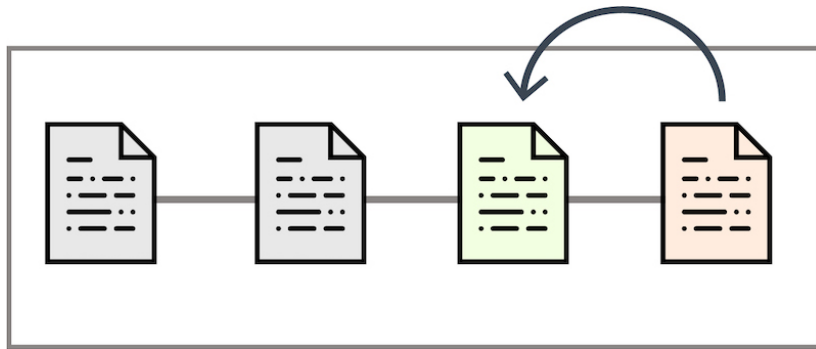


Git est un logiciel de gestion de versions décentralisé open-source créé par **Linus Torvalds** en 2005

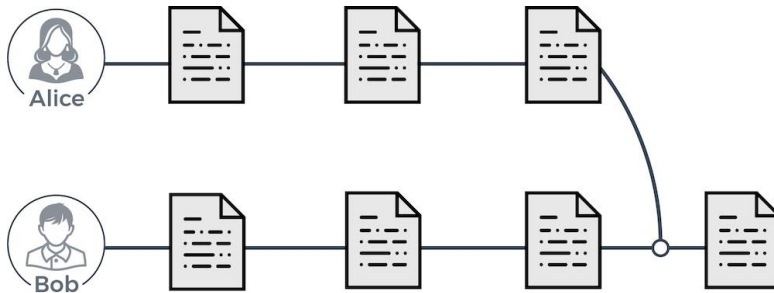
Pourquoi utiliser GIT ?

- Je vais travailler sur le fichier A et toi le B, mais ne touche pas au fichier A !
- J'ai le code du projet sur mon PC, personne n'y touche !
- J'ai effacé le code que je fait hier par erreur.
- Travailler en équipe sur la même application c'est pas pratique
- Qui a rajouté cette fonctionnalité dans l'application ?

Git



Je peux revenir sur une ancienne version
Et mettre de coté mes modifications pour la suite



Je peux réunir 2 codes facilement : **merge**

GitHub

GitHub est un service web d'hébergement utilisant le logiciel **GIT**



- fonctionnalités identique aux réseaux sociaux
- possibilité de suivre des personnes
- possibilité de récupérer rapidement des projets public avec ou sans GIT
- possibilité héberger un site web



- Pas de mot de passe , login , de clefs dans son repo
- Bien configurer son fichier **.gitignore**

GitLab

GitLab est un service web d'hébergement utilisant le logiciel **GIT**

GitLab offre des fonctionnalités en plus
Orientées devOps



Créez un compte GitHub

Créez votre propre dépôt ou repository

Un repository c'est une boîte qui va contenir notre projet

A partir de votre tableau de bord cliquer sur **New**

 Overview  Repositories 2  Projects  Packages

Type ▾

Language ▾

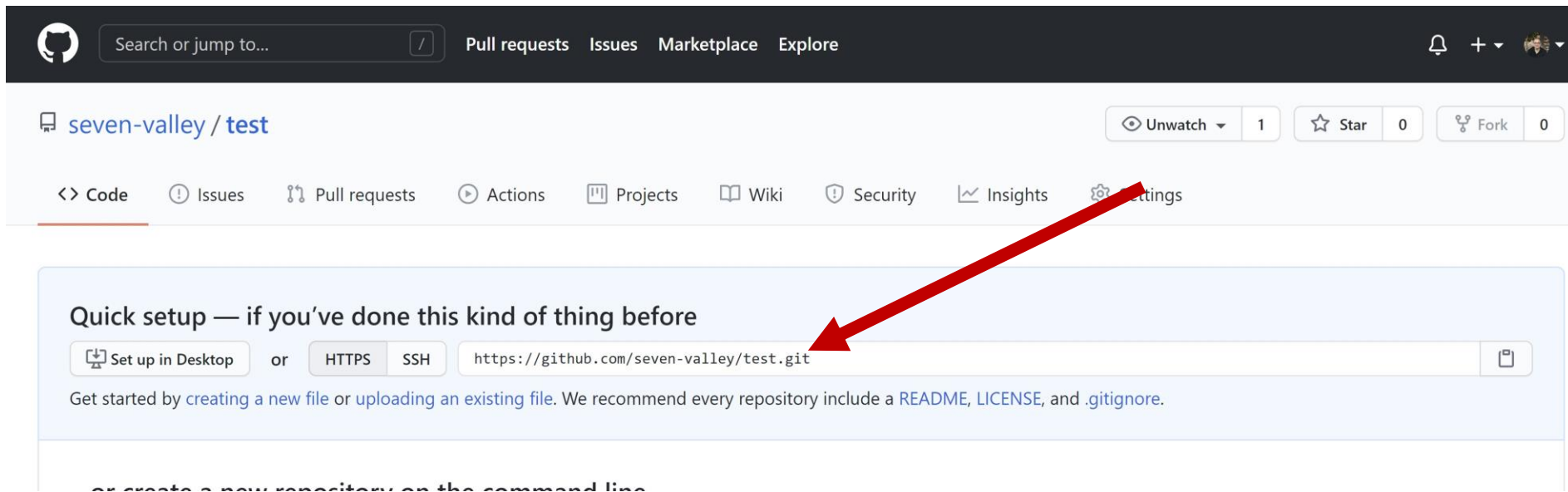
Sort ▾

 New



Créez un compte GitHub 2

Mémoriser votre adresse git de votre repo



The screenshot shows the GitHub interface for a repository named 'seven-valley/test'. The top navigation bar includes the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. The repository header shows the name 'seven-valley / test' and buttons for Unwatch (1), Star (0), and Fork (0). Below the header is a tabbed interface with 'Code' selected. The 'Quick setup' section is highlighted, showing options to 'Set up in Desktop', 'HTTPS', or 'SSH'. The repository URL 'https://github.com/seven-valley/test.git' is displayed in a text box, with a red arrow pointing to it. Below the text box, there is a note: 'Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.'

seven-valley / test

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH `https://github.com/seven-valley/test.git`

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

or create a new repository on the command line

Installation de GIT

<https://git-scm.com/downloads>



Avec un terminal sympa **cmdr mini**:

<https://cmdr.net/>



Ouvrir cmdr Mini

Création du répertoire pour nos projets GIT

```
cd\  
md projet-git  
cd projet-git  
  
md projet-demo  
cd projet-demo
```

Installation de GIT sur mon projet

Installation de git sur notre **projet-demo**

1.Je viens relier avec mon email et mon nom pour signer chaque commit

```
git config --global user.name "Seven Valley"
```

```
git config --global user.email augure@gmail.com
```

Installation de GIT sur mon projet

Installation de git sur notre **projet-demo**

1.Je viens relier avec mon email de compte GitHub

```
git config --global user.name "Seven Valley"  
git config --global user.email augure@gmail.com
```

2.J'installe git : création d'un repertoire **.git**

```
git init
```

Installation de GIT sur mon projet

3. Je définie le nom de ma branche **main**

```
git branch -M main
```

4. Je relie mon projet au repo de gitHub

```
git remote add origin https://github.com/lorem.git
```

Installation de GIT sur mon projet

5.J'ajoute tous mes fichiers

```
git add .
```

7.Mon premier commit

```
git commit -m "mon premier commit"
```

Je prends une "photo" de mon code a l'instant **t**

8.Et enfin je push vers gitHub

```
git push origin main
```

Une fois GIT installé

1. J'ajoute mes modifications

```
git add .
```

2. Je crée un commit

```
git commit -m "mon 2eme commit"
```

3. Et enfin je push vers **gitHub**

```
git push origin main
```

Commit et Push ??

