

Démarrer avec git

Guide Complet

Mise à jour du 27 février 2025

Git est un système de gestion de versions incontournable pour les développeurs.

Il permet :

- de **suivre** l'évolution d'un projet,
- de **collaborer** efficacement en équipe,
- de **gérer** différentes versions du code.

Notre objectif

- **Optimiser** notre workflow de développement,
- **Assurer** un suivi précis des modifications,
- **Faciliter** la gestion des branches.



Documentation

Site officiel

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

Documentation

<https://git-scm.com/book/fr/v2>

Installation sous Ubuntu

```
# Installation
sudo apt-get --yes install git

# Désinstallation
sudo apt-get --yes purge git
```

Configuration

```
# Voir la configuration
git config --list

# Configurer le compte par défaut
git config --global user.email "user_email"
git config --global user.name "user_name"

# Configure l'utilisation de notepad
git config --global core.editor "notepad"
```

Premier dépôt

Création du premier repo

```
# Création du fichier README
echo "# essai" >> README.md

# Initialisation du dépôt
git init

# Ajoute le fichier README à l'index
git add README.md

# Ajoute les derniers fichiers modifiés
git add *

# Commit
git commit -m "first commit"

# Remote sans identifiant
git remote add origin https://github.com/user-name/example.git

# Remote avec identifiant
git remote add origin https://user-name:user-password@github.com/user-name/example.git

# Push
git push -u origin master

# Pull
git pull
```

Commandes

```
# Initialiser le projet en dépôt git
git init

git config

# Configurer le compte par défaut
git config --global user.email "user_email"
git config --global user.name "user_name"

# Annuler le dernier commit
git commit --amend

# Vérifier l'état du projet
git status
```

```
# Envoie les fichiers dans le stage
git add *

# Verifie les différences
git diff

# envoie un commit
git commit -m "feature01"

# fait un add et envoie un commit
git commit -a -m "feature01"

# Liste les commits avec un éditeur
git log

# Liste les commits sur une ligne
git log --oneline

# Liste les commits détaillés sur une ligne sur un fichier
git log --oneline -p index.html

git push

# se positionne sur l'état au moment du commit id
git checkout id

# se repositionne sur l'état actuel
# git checkout master

# recupere un fichier à un moment et fait un commit
git checkout id index.html
git commit -a -m "nouveau commit"

# annuler un commit
git revert id

# Annuler mise en stage
git reset HEAD fichier
git merge
```

```
# Crée une branche prototype
git branch prototype

# Liste des branches
git branch

# Sélection d'une branche
git checkout prototype

# Sélection de la branche master
git checkout master
```