

# GIT

`git --version`

`git config`

`git help -a`

`git status`

Pour connaître le statut

`git config --global user.name "Vannina MI"`

`git config --global user.email "vannina.mi@hotmail.fr"`

`git init`

activer un dossier

`git add .`

ajoute tous les fichiers du répertoire courant

`git add nomdufichier.txt`

ajoute uniquement le fichier spécifié

`git commit -m "commentaireDuCommit"`

`-m` enregistre toutes les modifications effectuées

`git -a -m "commentaireDuCommit"`

**-a** demande de mettre à jour les fichiers existants dans son index

### **git log**

affiche l'historique :  
son SHA (IDENTIFIANT HASH)  
auteur  
date  
description

### **gitk**

affiche git log avec une interface graphique plus sympa  
pwd

**git checkout** SHADucommit  
pour se positionner sur un commit

git checkout master  
revenir sur le commit le plus récent ou revenir à la branche principale

**git revert** SHADucommit  
créer un nouveau commit qui inverse le précédent

**git commit --amend -m "nouveau commentaire"**  
modifier le message du dernier commit mais uniquement si je n'ai pas pushé le commit sur l'origin

## **git reset --hard**

Annuler tous les changements pas encore commits

## **remote**

sauv sur une autre machine : interne envoi sur un autre ordi ou externe : sur GitHub

git remote add origin URL

git remote add origin

<https://github.com/user/repo.git>

git remote add origin

<https://github.com/vannina/Exercice/ExoGit2.git>

## **git remote -v**

vérifier le nouveau remote

création du code => commit => remote

remote = push

## **GitHub :**

service en ligne

héberger ses repositories

outil gratuit

code open source

communauté de dev

partager morceaux code : gists : public ou secret

recupérer code d'un autre repo

git clone lienFourniParGitHub

Récupérer code d'un autre repository : clonage repository

Lien SSH ou HTTPS ou Subversion

copier l'url par ex HTTPS ↵ droite dans le dossier créer

copier le code suite au clonage

Création repository open source

"New repository"

(nom, description, statut : public ou privé)

Cocher case "initialise with a README" pour cloner le repo sur notre machine si pas encore créer

## option **git clone**

### 1 option HTTPS

plus facile à utiliser (identifiant + mdp)

### 2 options SSH

Il faut créer une clé SSH

Ajouter à ssh-agent

Ajouter une nouvelle clé SSH au compte GitHub

Tester la connexion SSH

Sécuriser ses clés SSH et config agent authentication

## **git push origin master**

Envoyer dans la branche master de mon remote origin

Le code est pushé consultable sur le repo GitHub

Tous les commits sont visibles et cliquables sur GitHub

### **git pull origin master**

Récupérer sur la branche master les mods (autres dev ou autres machines) en local sur repo en ligne

branche master = branche principale

### **git master**

### **git branch**

Voir les différentes branches du projet

### **git branch nvelle-branche**

créer une nouvelle branche

### **git checkout nvelle-branche**

se positionner dans une autre branche du projet (checkout)

### **git checkout -b nvelle-branche**

créer une nouvelle branche + s'y positionner

### **git branch -d ma-branche**

Supprimer une branche dans Git

### **git merge brancheA**

### **git merge brancheB**

fusionner des branches

**git checkout master**

**git merge branche 2aire**

Fusionner branche dans le master

Attention aux conflits fréquents

Parfois besoin de modifier le fichier si les 2 sont différents

**git commit**

pour comparer 2 versions de code

outil à télécharger

**git mergetool** nomOutil

**git blame nomduchichier.extension**

Permet de trouver une modification donnée

liste toutes les modifications avec SHADucommit

Puis faire un **git log** (rechercher le commit par le sha)

ou **git show** débutdusha (détails du commit)

Toutes les infos sensibles doivent être exclues du fichier sur github

Créer un fichier **.gitignore** (variables de configuration, clés APIs, autres clés, mdp...)

Ajouter le fichier dans git et le commiter aussi

**git stash**

sauvegarder son travail pas fini pour le reprendre plus tard sans commiter

## **git stash pop**

pour reprendre son travail mis en attente  
.pop vide le stash précédent (soit commiter ou soit refaire un stash)

Si besoin de garder mes modifs dans mon stash

## **git stash apply**

## **pull request ou (PR)**

Proposer une modif sur un projet open source

Récupérer le repo pour contribuer

Faire un fork du repo (faire une copie sur mon compte)

Cliquer sur fork

Cloner le projet

Copier URL et sur mon compte la coller dans le terminal

```
git clone https://github/....
```

Faire les modifs, se placer dans le repo créé, créer une branche et effectuer les modifs

```
git checkout -b my-new-feature
```

```
git commit -m "Added etc..."
```

Pusher la nvelle branche sur github

```
git push origin my-new-feature
```

Puis se placer sur son fork github sur la nouvelle branche et transmettre la demande

Cliquer "compare & pull request"

Rédiger un message explicatif à l'auteur