

# Utilisation de base de git

Samuel Ben Hamou

10 juillet 2020

## Résumé

Un petit guide pour le total néophyte que je suis avec git. J'y recense des GUI d'intérêt général ainsi que les commandes que l'on m'a présentées comme étant les plus utiles. Enfin, quelques liens utiles vers des cheat sheets provenant directement de GitHub.

## 1 GUI utiles

- `git gui` permet d'éditer les nouveaux commits et, tant que la branche courante n'a pas été mise à jour sur le serveur, il permet aussi de corriger les anciens commits.
- `gitk` permet de visualiser les branches d'édition, et éventuellement de gérer des commits simples.

## 2 Lignes de commande

### 2.1 Commandes d'utilité générale

- `git branch [branch name]` crée une nouvelle branche. Pour accéder à cette nouvelle branche (ce n'est pas le cas par défaut), utiliser `git checkout [branch name]`.
- `git branch` affiche les branches du projet courant.
- `git status` affiche l'historique des modifications du projet et indique s'il est à jour par rapport au serveur.
- `git gc` permet d'effacer définitivement les fichiers et branches inutiles.

### 2.2 Interaction avec le serveur

- `git clone [deposit name] [destination]` permet d'importer un projet.
- `git push` enregistre les modifications locales sur le serveur *pour la branche courante seulement*.

- `git pull` permet de mettre à jour le projet local depuis le serveur.

### 2.3 Gestion des commit

- `git commit -a -m "text"` committe la dernière action effectuée avec `text`.
- `git revert [pointer]` pose un anti-commit (c'est à dire qu'il rajoute un commit sur une action déjà committée pour annuler les changements du commit pointé). Le pointeur `[HEAD^]` correspond au dernier commit.
- `git reset [pointer]` fait la même chose que `git reset` mais en modifiant le commit pointé.
- `git checkout -f [file]` permet d'annuler les actions effectuées sur le fichier `[file]` depuis le dernier commit.

## 3 Liens utiles

**Cheat sheet 1.** Très générale, sur les branches et les commit : <https://github.github.com/training-kit/downloads/github-git-cheat-sheet.pdf>

**Cheat sheet 2.** Un peu plus complète : <https://education.github.com/git-cheat-sheet-education.pdf>

**Cheat sheet 3.** Une troisième, qui ressemble à la deuxième : <https://github.github.com/training-kit/downloads/fr/github-git-cheat-sheet.pdf>