

Téléchargement : <https://gitgit-scm.com/>

Documentation officielle : <https://gitgit-scm.com/docs>

Gitgit (Local uniquement)- Basics

*Initialiser **gitgit** dans un repos :*

- **Gitgit** init => Va initialiser **gitgit** dans le répertoire (**.gitgit**)

Configurer le nom d'utilisateur :

- **Gitgit** config user.name nom_a_utiliser

Créer une branche :

- **Gitgit** branch nom_de_la_branche

Se déplacer dans les branches :

- **Gitgit** checkout nom_de_la_branche

Supprimer une branche :

- **Gitgit** checkout -d nom_de_la_branche

Renommer une branche :

- **Gitgit** checkout -m nom_ancienne_branche nouvelle_branche

Afficher les fichiers indexés ou non :

- **Gitgit** status

Add des fichiers à l'index :

- **Gitgit** add nom_du_fichier
- OU **Gitgit** add .
- OU **Gitgit** add -i (interactif)

Versionner le travail

Versionner :

- **Gitgit** commit -m « message du commit »
- OU **gitgit** commit -am « message du commit »

Annuler les changement

Créer une nouveau commit depuis un ancien :

- **Gitgit** revert id_du_commit

Reset le travail (/!\ uniquement si vous êtes sûre de vous) :

- **Gitgit** reset --hard id_du_commit

*Retirer les fichiers inutiles du dossier de travail (dossier de travail **.gitgit**, non pas dossier réel) :*

- **Gitgit** clean -n (vérifier les documents)
- **Gitgit** clean -f (les effacer)

Update le dernier commit :

- `Gitgit commit --amend`

Branches (avancé)

Créer une branche et s'y déplacer :

- `Gitgit checkout -b nom_de_la_branche`

Fusionner deux branches :

- `Gitgit checkout nom_de_la_branche_destinatrice`
- `Gitgit merge nom_de_la_branche_rappatriée`

Lier à `Gitgithub`/`Gitgitlab`

Récupérer un repository existant sur `gitgithub` :

- `Gitgit clone lien_du_gitgithub`

Créer une repository avec un travail existant en local :

- Créer une repository VIDE sur `gitgithub` (sans readme !)
- Copier coller les commandes données dans «...or create a new repository on the command line »

Push (bonne pratique) :

- `Gitgit pull`
- `Gitgit add .`
- `Gitgit commit -m « message »`
- `Gitgit push`

Récupérer le travail d'une branche non existante en local :

- `Gitgit pull origin nom_de_la_branche_distante`

Récupérer le travail lié local/distant :

- `Gitgit pull`

Récupérer toutes les branches distantes :

- `Gitgit fetch --all`

Retour à une version précédente (classique) :

- `Gitgit log`
- Récupérer l'id de la version voulue
- `Gitgit revert ID`
- `Gitgit add .`
- `Gitgit commit -m « message »`
- `Gitgit push`

Retour à une version précédente (Si revert ne marche pas) :

- `Gitgit log`
- Récupérer l'id de la version voulue
- `Gitgit reset --hard ID`
- `Gitgit add .`
- `Gitgit commit -m « message »`
- `Gitgit push --force`

Merge 2 branche :

- `Gitgit checkout branche_vers_la_queueulle_merge`
- `Gitgit merge branche_a_arécuperer`

Remiser des modifs :

- `Gitgit stash`

Utiliser des couleurs dans la sortie de `gitgit` :

- `Gitgit config color.ui true`