

Configuration

```
git config --global user.name "nom"  
git config --global user.email "prenom.nom@cnrs.fr"  
git config --global color.ui auto  
git config --global credential.helper cache  
git config --global http.postBuffer 524288000  
• Vérifier la configuration  
git config --global -l
```

Créer un dépôt

- Créer un dépôt vide
`git init projet`
- Créer un dépôt en clonant un dépôt distant
`git clone url`
- Créer un dépôt local dans un répertoire local existant
`cd repertoire_projet
git init git add -A`

Ignorer des fichiers

- Créer la liste des fichiers à ignorer et la publier
`git config --global core.excludefiles **/*.*log
Modifier le fichier .gitignore
git add .gitignore
git commit -m "Partage des fichiers à ignorer"`
- Afficher la liste de tous les fichiers ignorés
`git ls-files --other --ignored --exclude-standard`

Historique

- Afficher tous les commits (format par défaut ou court)
`git log
git log --pretty=short`
- Afficher les *x* derniers commits
`git log -n x`
- Afficher les commits d'un fichier ou d'un répertoire
`git log fichier
git log repertoire/`
- Afficher des statistiques pour chaque fichier modifié
`git log --stat`
- Afficher le contenu d'un commit
`git show id_commit`

Modifications locales

- Annuler les modifications réalisées dans un fichier
`git checkout -- fichier
git reset [--mixed] HEAD fichier`
- Ajouter des fichiers au prochain commit
`git add fichier1 fichier2 fichier3`
- Enlever un fichier du prochain commit
`git rm --cached fichier`
- Supprimer un fichier
`git rm fichier`
- Supprimer récursivement les fichiers d'un répertoire
`git rm repertoire/ -r`
- Renommer un fichier
`git mv fichier nouveau_nom`
- Déplacer un fichier
`git mv fichier destination/`
- Afficher l'état des fichiers nouveaux ou modifiés
`git status`
- Afficher les modifications des fichiers suivis modifiés
`git diff`
- Afficher les modifications du prochain commit
`git diff --cached`
- Effectuer un commit
`git commit
git commit -a (ajouter automatiquement les fichiers)
git commit -m "Message du commit"`
- Modifier le dernier commit
`git commit --amend`
- Etiqueter le dernier commit
`git tag nom_tag`
- Annuler les *n* derniers commit
`git revert HEAD (dernier commit)
git revert HEAD~ (2 derniers commit)
git revert HEAD~2 (3 derniers commit)`
- Retourner à la version du dernier commit
(Supprime les nouveaux fichiers et les modifications)
ATTENTION : Cette opération ne peut pas être annulée
`git reset --hard HEAD`

Branches

- Afficher la liste des branches
`git branch`
- Créer une nouvelle branche
`git branch nom_branch`
- Basculer sur une branche
`git checkout nom_branch`
- Fusionner une branche dans la branche courante
`git merge nom_branch`
- Supprimer localement une branche
`git branch -d nom_branch`
- Afficher les différences entre deux branches
`git diff nom_branch1...nom_branch2`

Dépôts distants

- Afficher la liste des dépôts déclarés
`git remote -v`
- Afficher des informations sur un dépôt
`git remote show nom_depot (ex:origin)`
- Déclarer un dépôt
`git remote add chemin|url`
- Déclarer le dépôt origin
`git remote add origin url`
- Récupérer les données d'un dépôt déclaré
`git fetch nom_depot`
- Récupérer les données de la branche d'un dépôt et fusionner dans la branche courante
`git pull [nom_depot] [nom_branch_distant]`
- Publier les modifications locales d'une branche
`git push [nom_depot] [nom_branch_locale]`
- Supprimer une branche dans un dépôt déclaré
`git push nom_depot :nom_branch_distant`
- Publier nom_depot --delete nom_branch_distant
- Publier les informations de tags
`git push nom_depot --tags`