

# Découverte de git

## Travail en local

1. Créez un nouveau répertoire pour y mettre le résultat de votre travail dans le cadre de ce sujet, puis faites-en votre répertoire courant dans votre terminal.
2. Exécutez la commande `ls -la` pour constater que ce répertoire est bien vide.
3. Configurez vos informations personnelles :

- `git config --global user.name "Prénom Nom"`
- `git config --global user.email "prenom.nom@etu.unistra.fr"`

4. Initialisez un **dépôt** git avec `git init` puis faites à nouveau `ls -la` pour voir les informations ajoutées par git.
5. Créez un fichier `README.md` et écrivez-y quelques lignes de prose.
6. Affichez l'état actuel du dépôt avec `git status`.
7. Avec `git add README.md` ajoutez vos changements à l'*index*.
8. Faites à nouveau `git status` pour constater le changement d'état.
9. Enregistrez vos changements dans une **révision** avec `git commit`.
10. Faites à nouveau `git status` pour constater le changement d'état.
11. Créez une branche avec `git branch amendements`.
12. Basculez sur cette branche avec `git checkout amendements`.
13. Modifiez le fichier et créez plusieurs révisions supplémentaires.
14. Consultez l'historique avec `git log`, retournez sur la branche principale avec `git checkout master` puis consultez à nouveau l'historique.
15. Fusionnez la branche `amendements` avec la branche `master` via la commande `git merge amendements master`.

# Dépôt distant

Vous aurez besoin d'une clé SSH. Vérifiez si vous en avez déjà une (clé privée `id_rsa` et clé publique `id_rsa.pub` dans `~/.ssh/`) et si ce n'est pas le cas créez-en une avec la commande `ssh-keygen`. Les réponses entre crochets sont les valeurs par défaut : faites simplement directement entrée pour les conserver.

1. Rendez vous sur l'instance Gitlab de l'Université (<https://git.unistra.fr/>), et connectez vous. Allez dans Paramètres, Clé SSH et ajoutez votre clé **publique**.

2. Créez un nouveau projet **privé** que vous nommerez `TechDev-TP1-git-nomprenom`.

⚠ Attention à **ne pas cocher la case** pour initialiser le dépôt.

3. Récupérez l'adresse du dépôt en cliquant sur le bouton Clone. Assurez vous de prendre celle correspondant au protocole SSH.

4. Associez le **dépôt distant** de ce projet à votre **dépôt local** : dans votre terminal, lancez `git remote add origin adresse_du_depot`

5. Envoyez votre historique local sur Gitlab avec `git push` puis affichez le dans l'interface web.

6. Quelles branches sont présentes dans votre dépôt local ? Et dans le dépôt distant ? Comment les aligner ?

# Collaboration

Trouvez un camarade avec lequel faire cette partie. Il faudra la faire deux fois pour que chacun puisse endosser les deux rôles.

1. L'étudiant A ajoute l'étudiant B à son projet avec le rôle *Reporter*.

2. L'étudiant B créer un *fork* du projet, le clone, y apporte une modification qu'il pousse dans son *fork* puis ouvre une *merge request*.

3. L'étudiant A passe en revue les changements et les intègre via l'interface web.

4. L'étudiant A récupère en local ces changements.

5. Dessinez un schéma avec les quatre dépôts et des flèches annotées pour chacune des étapes que vous avez effectuées.