

## PRÉPARER SON ENVIRONNEMENT POUR UN TP

Ce document explique ce qu'est Git et son utilisation. Pour ceux qui utilisent Mac, vous trouverez un guide afin de bien installer Git sur celui-ci puisqu'il ne vient pas de base. Finalement la dernière section détail comment bien cloner le répertoire dans lequel vous effectuerez vos travaux pratiques.

### NOTIONS DE GIT

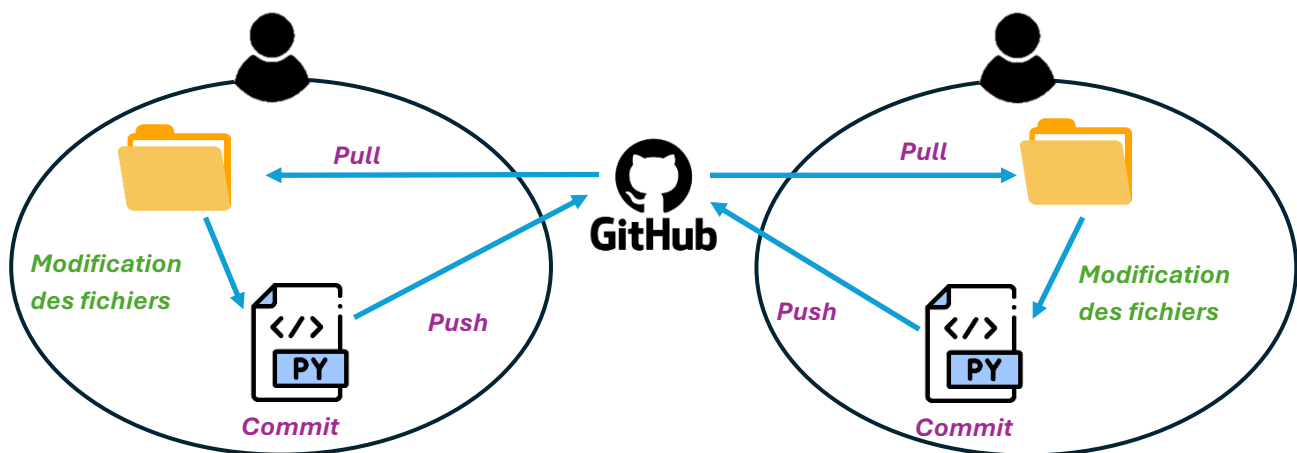
Git est un outil qui permet de suivre les modifications d'un projet, souvent du code, comme un journal de bord collaboratif. Il aide les développeurs à travailler ensemble sans écraser le travail des autres.

- **Update Project** (pull): Si quelqu'un d'autre a fait des changements sur le projet, tu peux les récupérer sur ton ordinateur avec un "pull". Ça te permet de rester à jour avec les dernières modifications.

- **Commit** : C'est comme prendre une photo de ton travail à un moment donné. Tu enregistres les changements que tu as faits avec un petit message pour expliquer ce que tu as modifié.

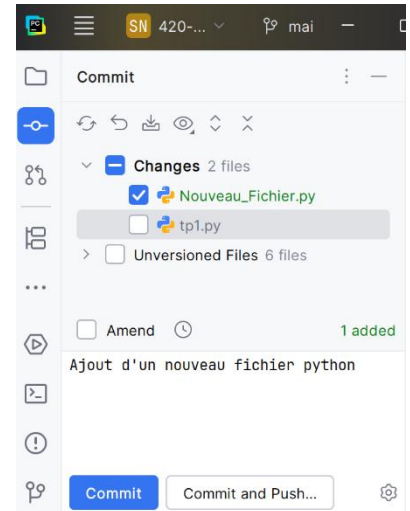
- **Push** : Une fois que tu as fait des commits sur ton ordinateur, tu peux les envoyer vers un serveur (comme GitHub) pour que les autres puissent voir ton travail. C'est ça, "pusher".

En résumé : tu commit pour enregistrer, tu push pour partager, tu pull pour récupérer. Git garde tout bien organisé, même quand plusieurs personnes travaillent en même temps.



## Étapes pour bien utiliser Git

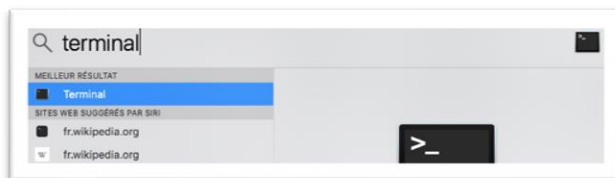
1. Ouvrir le projet, directement vérifier que nous sommes à jour en faisant **Update Project** (pull).
2. Effectuer les modifications voulues.
3. Vérifier qu'aucune modification n'a été faite entre temps avec **Update Project**.
4. Pousser les changements dans le répertoire GitHub :
  - a. Appuyer sur l'option **Commit**.
  - b. Sous **Changes**, vous verrez tous les fichiers que vous avez modifiés. Sélectionner les fichiers que vous souhaitez envoyer sur le répertoire.
  - c. Dans la boîte de texte en bas, écrivez un commentaire qui représente les modifications que vous avez apportées.
  - d. Si vous souhaitez faire d'autres modifications avant de les envoyer à vos collègues vous pouvez simplement faire **Commit** et faire **Push** plus tard. Sinon vous pouvez directement les envoyer sur le répertoire en appuyant sur **Commit and Push**.



## INSTALLER GIT SUR MAC

Pour installer Git sur Mac il faut d'abord installer HomeBrew puis nous pourrons installer Git. Les étapes ci-dessous donnent la marche à suivre :

1. Ouvrir le module de recherche de Apple en appuyant sur la touche **Command + Espace**.
2. Entrer dans la barre de recherche le mot « **Terminal** » puis appuyant sur **Enter**.



3. Copier et coller la ligne suivante dans le terminal puis appuyer sur **Enter**. Ceci débutera l'installation de HomeBrew :

**/bin/bash -c "\$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"**



4. Taper votre mot de passe de votre ordinateur. Il est normal de ne rien voir s'afficher lorsque vous le tapez, il s'agit d'une mesure de sécurité d'Appel. Appuyer sur **Enter**.
5. Appuyer sur **Enter** pour confirmer l'installation (cela prendra quelques minutes).

```
Press RETURN/ENTER to continue or any other key to abort:
```

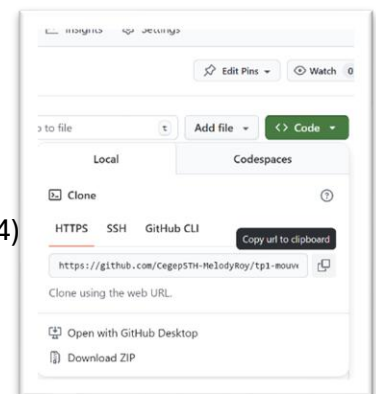
6. Une fois que l'installation est terminée, tapez la ligne de commande suivante dans le terminal : **brew install git**

```
MacBook-Air-de-Kelly-Anne:~ kelly-anne$ brew install git
```

7. Une fois que l'installation est terminée, redémarrez votre ordinateur.
8. Vous pouvez maintenant utiliser Git.

## CLONER UN TP DE GITHUB CLASSROOM SUR PYCHARM

1. Cliquez sur le lien GitHub Classroom distribué dans la section Documents de Léa.
2. Créer ou rejoindre une équipe de 2 ou 3 personnes.
  - a. Un premier membre de l'équipe doit créer une équipe. Nommez-la en utilisant tout les DA des futurs membres.
  - b. Les autres membres pourront parcourir les équipes existantes et rejoindre à leur équipe.
3. Accéder à votre Repository d'équipe et cliquer sur le bouton vert **Code**.
4. Copier le lien HTTPS
5. Dans PyCharm cliquer sur **File** puis **Project from version control**
6. Dans la case URL, coller l'url HTTPS de GitHub Classroom (étape 4) puis appuyer sur clone
7. Sélectionner l'option **Use Token...**
8. Appuyer sur **Generate** ce qui vous redirigera sur GitHub
9. Authentifier vous.
10. Modifier l'expiration pour **No expiration**.
11. Au bas de la page, appuyez sur **Generate Token**
12. Copier le token créé
13. Revenez sur PyCharm et coller le token puis appuyer sur **Log In**
14. Le projet pourra alors être clonée



Expiration

 No expiration ▼