



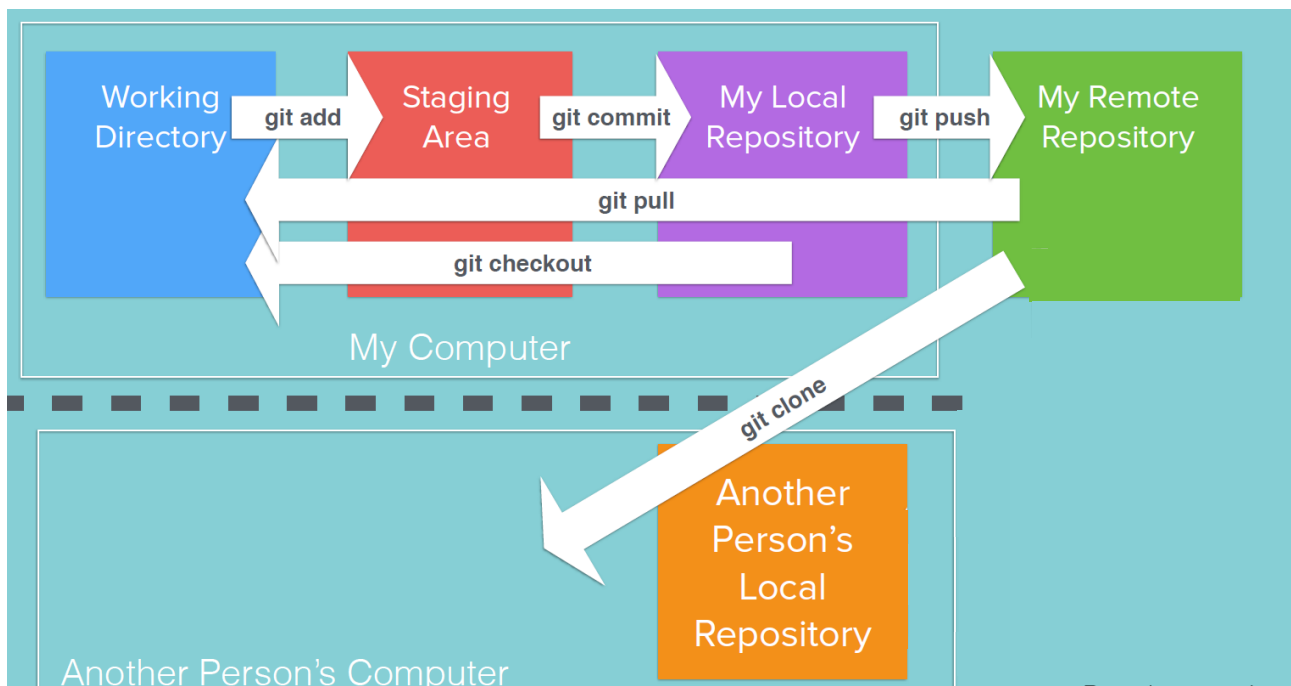
# git

# &



## C'EST QUOI ?

GIT EST UN SYSTÈME DE GESTION DE VERSIONS DE FICHIERS DÉCENTRALISÉ PERMETTANT DE GARDER EN MÉMOIRE TOUTES LES MODIFICATIONS APPORTÉES À DES FICHIERS. IL EST DONC PRATIQUE POUR LA GESTION D'UN PROJET EN GROUPE ET LE DÉVELOPPEMENT DE TACHES TRANSVERSALES.



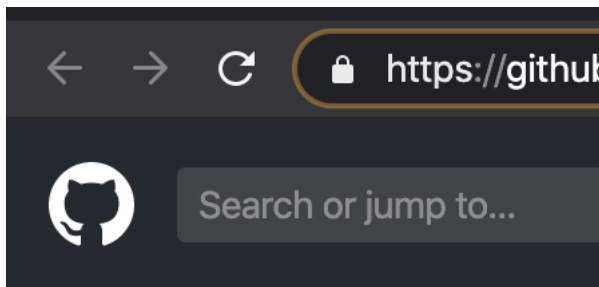
# Rejoindre le repository sur Git Hub 🐙

- Go to -> <https://github.com/>

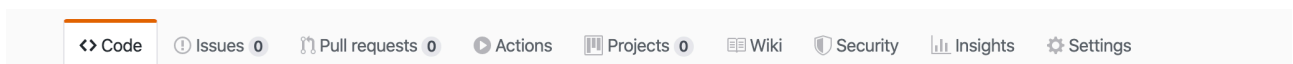
---

## Rejoindre un Repository :

- Cliquer sur le chat :



- Sélectionner le Repository. (ex : Machin/unProjet)
- Ceci est la barre principal de navigation dans le projet :



## Utiliser les commandes Linux de base 🐧

```
-pwd, cd, cd .., cd ../, ls, ls -l, mkdir, touch, cat, vim
```

(navigation, création, visualisation, modification)

```
rm
```

(Suppression)

```
man
```

```
man ls
```

(manuel)

```
:wq ou x
```

(quitter vim)

# Utiliser les commandes GIT 🐱

---

Naviguer dans le repository :

```
git clone https://github.com/Zolpra/coursGit.git
```

(cloner le repository en local)

```
git branch
```

->

```
macbook-pro-de-alexandre-2:coursGit alexandredouillard$ git branch
* master
macbook-pro-de-alexandre-2:coursGit alexandredouillard$
```

(Connaitre la branche actuelle)

```
git checkout
```

(Changer de branche)

```
git checkout -b uneBranche
```

(Tirer une nouvelle branche depuis la branche actuelle et checkout dedans)

---

Envoyer des modifications :

```
git status
```

(Connaître les fichiers modifiés/créés et prêts pour l'envoi)

2 types d'affichage :

```
macbook-pro-de-alexandre-2:coursGit alexandredouillard$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

    new file:   test.txt

macbook-pro-de-alexandre-2:coursGit alexandredouillard$
```

On branch master

No commits yet

Untracked files:

(use "git add <file>..." to include in what will be committed)

test.txt

nothing added to commit but untracked files present (use "git add" to track)  
macbook-pro-de-alexandre-2:coursGit alexandredouillard\$

```
git add test.txt
```

(ajouter un nouveau fichier)

```
git add -p
```

(ajouter des modifications)

```
git commit -m "first commit"
```

(assigner une référence de commit à l'envoi de toutes les modifications/créations)

*Ce qui donne :*

```
macbook-pro-de-alexandre-2:coursGit alexandredouillard$ git commit -m "first commit"
[master (root-commit) 6157610] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 test.txt
macbook-pro-de-alexandre-2:coursGit alexandredouillard$
```

```
git push origin uneBranche
```

(envoyer les modifications)

*Ce qui donne en cas de succès :*

```
macbook-pro-de-alexandre-2:coursGit alexandredouillard$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 216 bytes | 216.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/Zolpra/coursGit.git
 * [new branch]      master -> master
macbook-pro-de-alexandre-2:coursGit alexandredouillard$
```

```
git pull uneBranche
```

(mettre a jour une branche (download))

```
git fetch une branche
```

(se déplacer dans une branche déjà existante mais non présente en local)

```
git merge unebranche
```

(fusionner « uneBranche » avec la branche actuelle)