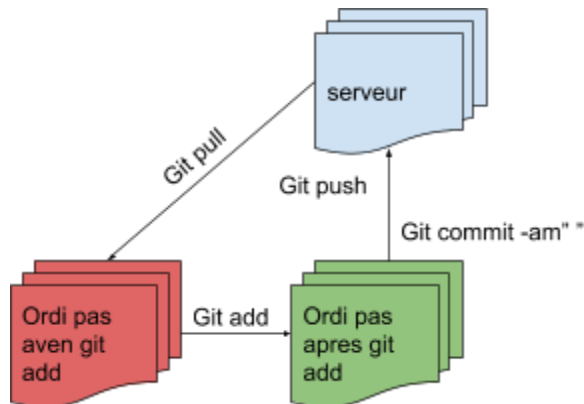


l'idée générale de git est : un dossier sur le serveur, et deux copies sur l'ordi mais on voit qu'un (le rouge, on n'a pas d'accès sur l'autre).



1. cd dans dossier projet dans le terminal.
  2. git pull pour mettre à jour ton version de dossier (si j'ai déjà changer quelque chose).
  3. fait ce que tu veux faire. (changer un fonction, ajouter une, ajouter un fichier, ...).
  4. git add le nouveau fichier (ou ce que t'as changer). ex: git add test.c
  5. commit -am" écrit une petite comment pour expliquer ce que t'as fait".
  6. git pull (si j'ai changer autre chose au meme temp pour que tu m'écrase pas mes changement).
  7. git push pour mettre ton changement en ligne.
- exemple dans mon projet ocaml :

1)

```
MacBook-Pro:ocaml kinan$ cd to projet file
```

2)

```
MacBook-Pro:ocaml kinan$ git pull
```

3)

```
MacBook-Pro:ocaml kinan$ edit or add file ex: myTest.ml
```

4)

```
MacBook-Pro:ocaml kinan$ git add myTest.ml
MacBook-Pro:ocaml kinan$
```

5)

```
MacBook-Pro:ocaml kinan$ git commit -am "ex: adding comments"
```

6)

```
MacBook-Pro:ocaml kinan$ git pull
```

```
Already up to date.
```

```
MacBook-Pro:ocaml kinan$ git add test.ml
```

7)

```
MacBook-Pro:ocaml kinan$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 1.66 KiB | 1.66 MiB/s, done.
Total 4 (delta 1), reused 0 (delta 0)
To http://moule.informatique.univ-paris-diderot.fr:8080/kinan/ocaml.git
665791c..db1262e master -> master
```

8) tu peux toujours vérifier l'état de ton dossier (si y'a des chose sur ton ordi mais pas sur le serveur ou l'inverse) git status.

```
MacBook-Pro:ocaml kinan$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
MacBook-Pro:ocaml kinan$
```