

Résumé des commandes de base de Git

> Git version

Donne la version de git installée.
Si il ne répond rien c'est que Git n'est pas installé.



```
$ git --version
```

```
$ git version 2.28.0.windows.1
```

> Git config

Configure votre identité et va permettre à Git d'identifier par qui les modifications ont été faites.




```
$ git config --global user.name "Lucas JULIAN"  
$ git config --global user.email "lucasjulian-esd@outlook.fr"  
$ git config -l
```

> Mkdir, cd, git init

Mkdir pour « Make directory » créer un dossier

Cd pour « Change directory » déplace votre ligne de commande dans le répertoire indiqué

Git init, initialise git dans le répertoire où vous vous trouvez.



```
$ mkdir new_project  
$ cd new_project  
$ git init
```

> Git status

Git status va lister les fichiers modifiés



```
$ git status
```

> Créer un commit

Créer un commit se déroule en 3 étapes:

Git `commit -message « votre message de commit »`: créer un commit avec un message décrivant ce commit



```
$ git status  
$ git add <mon_fichier>  
$ git commit --message "message de commit"
```

> Git diff

Git diff : liste les modification (ligne par ligne et non par fichier)



```
$ git diff  
$ git diff <un_fichier_spécifique>
```

> Git commit --amend

`Git commit --amend` : permet de mettre à jour un commit (son contenu), on le fait après avoir rajouter (`git add`) les nouvelles modifications



```
git commit --amend
```


> Le « Branchement »

Créer une branche (avec le nom indiqué)



```
$ git branch ma-branche
```

