WITZ Francesco
INFO1
LO43 – A19

Rôles des commandes Git

1. ## git config
   Cette commande permet de paramétrer les identifiants git tel que le nom d'utilisateur ou l'adresse email :

   $ git config --global user.name "NOM" -> nom de l'utilisateur
   $ git config --global user.email [Adresse mail] -> adresse mail liée à toute les opérations de commit

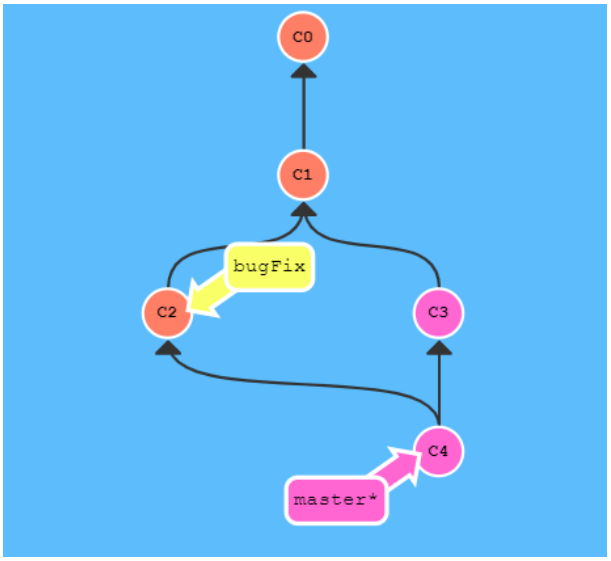
2. ## git init
   Cette commande permet à l'utilisateur d'initialiser un nouveau répertoire .git vide en local, et également de créer des sous-répertoires dans un répertoire existant.

   $ git init [nom-du-projet] -> Crée un dépôt local à partir du nom du projet

3. ## git status
   Cette commande indique l'état de chaque fichier dans le répertoire git, c'est-à-dire si ils ont été modifié et si ils sont à jour sur le répertoire distant ou non (si il faut effectuer un commit ou pas).

4. ## git merge
   Cette commande fusionne deux branches entre elle, en incorporant sur la branche courante le travail effectué sur la deuxième. En général, on merge sur le master (on met en production) les autres branches (le travail en développement) pour avoir un répertoire de travail propre.

   $ git merge bugFix (en étant sur master, c'est-à-dire après avoir fait $ git checkout master)



5. git diff
   Cette commande permet d'afficher les modifications entre l'arbre de travail et l'index du répertoire. C'est très utile pour visualiser ce qui a été modifier mais pas encore indexer.

6. git blame
   Cette commande permet de savoir quel contributeur du répertoire à effectuer la dernière modification, pour ensuite déterminer qui va se faire taper sur les doigts suite à l'apparition soudaine d'un bug. Plus sérieusement, cette commande est réellement utile pour faire des code reviews efficaces et demander aux personnes concernées de corriger les éventuelles erreurs engendrées.