**Github:**

**- Générer une clé SSH:** Dans le terminale→ ssh keygen -t ed25519 -C “adresse mail” → Suivre la suite→ Aller chercher la clé dans .ssh→ Copier la clé public→ Aller sur Github→ Créer une nouvelle clé→ Coller la clé

**- Git init :** A faire seulement sur le dossier racine

**- Retirer config Git d’un dossier:** rm -rf .git

- **Modifier URL origin:** git remote set-url orgin URL du nouveaux Repo

**- Envoyer un nouveau dossier local sur un nouveaux Repo:** Créer un Repo sans README→ Revenir au terminale et ce mettre au niveaux du dossier ou fichier à envoyé→ git init→ git add --all→ git commit -m “commits”→ git remote add origin ssh du repo→ git push origin main

**- Envoyer un dossier modifier local à son Repo Github:** git add –all (Pour remplacer tout les fichiers, sinon indiquer le fichier après add)→ git status (Vérifie le modifs)→ git commit -m “commits”→git push origin main

**- Mettre un fichier local à jour après modif sur Github:** git pull origin main

**- Installer git:** Installer le programme git→ Aller sur le terminale→ git config --global user.name “Nom d’User Github”→ git config --global user.email “email”→ git config --global pull.rebase false→ git config --global init.default Branch main

**- Ignorer des fichiers:** A la racine du dossier git→ Créer un dossier .gitignore→ nano .gitignore→ (#ignorer les fichiers .log) \*.log→ git add .gitignore→ nommé le fichier concerné en .log

**- Fusionner une branche:** Depuis main→ git pull origin main→ git merge nom\_de\_branch→

**- Gérer un conflit:** Editer le fichier en conflit et garder une des modifications → enregistrer→ fusioner.

**Commandes:**

- git check out.. = Récupère les fichier et dossier en l’état de la dernière MAJ

- git checkout -b nom\_de\_branch = Basculer et créer la branche

- git branch -d nom\_de\_branch = Supprime la branche

- git checkout main = Revenir sur le main

- git push origin nom\_de\_branch = Push sur la branche

- git init = rend un dossier classique en dossier locale

- git log = Affiche les commit du fichier

- git checkout N°commit = Lecture du commit

- git revert N°commit = Annule le commit

- git reset N°commit = Annule le commit et ceux qui suivent

- git reset N°commit --hard = Supprime aussi les commit locaux

- git commit --amend -m “commits” = annule le précédent commit et le remplace

- git stash = Sauvegarde les modif en local

- git stash apply = Appliquer les modifications