git init Para empezar un proyecto de git

git status -s Para ver el estado de los archivos del proyecto (?? Significa que no se está haciendo seguimiento del archivo, M significa que está modificado, A que está en el área de ensayo)

git add para subir un archivo al área de ensayo.

git add . para subir todos los archivos al área de ensayo.

git commit -m “comentario” para subir el archivo en el repositorio local.

git log --oneline permite conocer todas las modificaciones hechas.

git reset --hard [código del log] permite restaurar a una copia anterior del log.

git commit -am “comentario” con el -am se puede hacer el add y commit en un solo comando.

git commit --amend permite abrir el editor vm para modificar la descripción de un commit.

\*\*VM: al estar dentro del editor comenzamos escribiendo :i para editar los commits. Para salir del editor se coloca :wq

git tag {fecha}{versión} -m “comentario” el tag nos permite darle una versión a nuestro programa.

git push --tags se subirán los tags creadas.

crear rama

- git branch (nombre de la rama"No debe llevar ni paréntesis ni corchetes")

- git log --oneline o git branch (confirma visualmente la creación de la rama)

moverse de rama

- git checkout (nombre de la rama a la que se mueve “No debe llevar ni paréntesis ni corchetes")

- git branch (para confirmar la ubicación de la rama )

APARTIR DE AQUI, SE PUEDEN UTILIZAR LOS COMANDOS PARA AGREGAR VERSIONES

Moverse a master

- git checkout master

Unir ramas

- git merge (nombre de la rama a unir "No debe llevar ni paréntesis ni corchetes")

Eliminar ramas

-git branch -d nombreRama