

'COMANDOS GIT'

*git init: Inicializa un nuevo repositorio

*git clone: Clona un repositorio remoto a tu máquina local.

git clone <https://github.com/user/repo.git>

*git status: Muestra el estado de los archivos en el repositorio.

*git add: Añade archivos al área de preparación (staging area).

git add archivo.txt

*git commit: Guarda los cambios en el repositorio con un mensaje.

git commit -m "Mensaje del commit"

*git push: Sube los commits locales al repositorio remoto.

git push origin main

*git pull: Descarga cambios del repositorio remoto y los

fusiona.

git pull origin main

*git fetch: Descarga datos del repositorio remoto sin fusionarlos.

git fetch origin

*git merge: Fusiona ramas en la rama actual.

git merge feature-branch

*git branch: Lista, crea o elimina ramas.

git branch # Lista las ramas

git branch nueva-rama # Crea una nueva rama

git branch -d vieja-rama # Elimina una rama

*git checkout: Cambia de rama o restaura archivos.

git checkout main # Cambia a la rama main

git checkout -- archivo.txt # Restaurar

archivo.txt

*git switch: Cambia de rama (alternativa moderna a git checkout para ramas).

git switch main

*git log: Muestra el historial de commits.

*git diff: Muestra diferencias entre commits, ramas o archivos.

git diff HEAD | HEAD # Diferencia entre el último y penúltimo commit

*git reset: Deshace cambios en el área de preparación o el historial de commits.

git reset archivo.txt # Saca un archivo del área de preparación

git reset --hard HEAD | # Elimina el último commit

*git rm: Elimina archivos del repositorio y del área de trabajo.

git rm archivo.txt

*git stash: Almacena temporalmente

cambios sin hacer commit.

git stash

git stash pop # Recupera el stash guardado

*git rebase: Reaplica commits en una nueva base de rama.

git rebase main # Rebasea la rama actual con la rama main

*git tag: Crea o lista etiquetas en commits específicos.

git tag v1.0 # Crea una etiqueta en el último commit

git tag # Lista todas las etiquetas

*git remote: Administra conexiones a repositorios remotos.

git remote add origin

<https://github.com/user/repo.git> # Añade un remoto

git remote -v # Lista los remotos configurados

*git revert: Crea un nuevo commit que

deshace cambios de un commit anterior.
git revert HEAD # Revierte el último commit

* git show: Muestra detalles sobre un objeto Git (commit, etiqueta, etc.).
git show HEAD # Muestra detalles del último commit

* git blame: Muestra quién cambió cada línea de un archivo y cuándo.
git blame archivo.txt

* git cherry-pick: Aplica un commit específico de otra rama.
git cherry-pick abcd234 # Aplica el commit con el hash abcd234

* git bisect: Utiliza la búsqueda binaria para encontrar un commit defectuoso.
git bisect start
git bisect bad # Marca el commit actual como defectuoso
git bisect good abcd234 # Marca un commit conocido como

correcto

* git archive: Crea un archivo comprimido de una versión específica del repositorio.
git archive --format=zip HEAD >
archivo.zip

* git submodule: Administra submódulos dentro de un repositorio.
git submodule add
https://github.com/user/submodule.git
git submodule update --init # Inicializa los submódulos

* git config: Configura opciones de Git, como el nombre de usuario y correo electrónico.
git config --global user.name "Tu Nombre"
git config --global user.email
"tu.email@ejemplo.com"

* git clean: Elimina archivos no rastreados del directorio de trabajo.
git clean -f # Fuerza la eliminación de archivos no

rastreados

*git reflog: Muestra el historial de referencias de tus movimientos de HEAD.

*git mv: Mueve o renombra un archivo dentro del repositorio.

git mv archivo.txt nuevo-nombre.txt

*git ls-files: Lista archivos en el Índice de Git.

*git grep: Busca un patrón en los archivos versionados del repositorio.

git grep "buscar-este-texto"