

COMANDOS MAS UTILIZADOS

*git status: muestra el estatus de los archivos

esto muestra si ya fueron guardados o no.

*git add "nombre archivo": agrega el archivo nuevo al repositorio

*git add .: agrega todos.

Si hacemos nuevamente git status con todos los archivos subidos

este mostrara que todos fueron guardados

Una vez agreguemos un archivo o hagamos un cambio es bueno o necesario crear un commit.

*git commit -m "He subido un nuevo archivo": con esto podemos generar un punto en el tiempo en el cual se puede conocer que se hizo y cuando necesitemos volver con esta informacion nos basamos

*git log: genera un historial de lo que se hizo

Ahora si agregamos algo a los archivos que ya estan montados

cuando sacamos un git status nos va a aparecer modificado, si lo volvemos a comentar git commit -m "lo he cambiado" ahora cuando git status nos va a decir que estan todos guardados.

*rm -rf .git: borramos el Track del git, osea borra el seguimiento eliminando la etiqueta (main) del directorio.

*en algunas versiones la etiqueta (main) se llama (master).

si queremos cambiar este nombre se usa git branch -m master

con esto cambiamos el nombre de la rama principal, esto se hace porque en algunas versiones no funciona main.

*git restore --staged: regresa el estado de guardado o lo subido.

*git checkout "codigo del momento": te lleva a revision del primer momento del commit

*git log --oneline: resume drasticamente los momentos del tiempo.

*git checkout main: vuelve al momento actual

Como subir archivos al GIT HUB remoto:

solo se pueden subir comitiados

*git push origin master: sube los archivos (!Se debe de tener en cuenta que este Origin master puede cambiar de como esten nombrados

en la computadora, en este ejemplo a mi me toco poner "original main!").

*git clone URL: clona los archivos actualizados en el repositorio remoto, para ello

debemos ir a git hub y darle codigo - Clone - "Se copia la url de alli".

*para sincronizar archivos, ejemplo trabajas en un sitio y luego vas a tu casa y necesitas sincronizar para ello

usas esto, *git pull origin master (tener en cuenta que aveces puede tener otro nombre "origin master": cambia y modifica o actualiza lo que ya esta.

*git branch --list: enumera las ramas del repositorio

*git branch Nombre: en nombre es cualquier y crea una nueva rama con ese nombre

*git branch -d nombre: elimina la rama con ese nombre. Git trata de no eliminar lo que se ha mezclado

*git branch -D nombre: fuerza la eliminacion incluso si tiene cambios sin fusionar

*git checkout nombre_rama: para moverse entre las ramas, git solo permite que nos movamos a otra rama si no tenemos cambios

si tenemos cambios debemos o eliminar los cambios o confirmarlos con un commit.

*git push origin (branch) con esto traemos los cambios utilizados por la rama.