

Para crear un nuevo usuario:

- `git config --global user.name "nombre de usuario"`
- `git config --global user.email email@email.com`

Para crear un nuevo repositorio:

- `git init`

Para verificar el estado de un archivo o carpeta:

- `git status`

Agregar archivos:

- `git add [ubicación del archivo] (directorio/carpeta/archivo)`

Agregar todos los archivos:

- `git add .`

Realizar el commit de un archivo o directorio:

- `git commit [archivo o directorio] -m "nombre del commit"`

Remover:

- `rm [archivo] (remueve el archivo)`
- `rmdir [carpeta] (remueve una carpeta vacía)`
- `rm -r [carpeta] (remueve una carpeta incluyendo el contenido)`

Crear:

- `mkdir [carpeta] (crea una nueva carpeta)`
- `touch [archivo] (crea un nuevo archivo)`

Ver:

- `ls (muestra los archivos o carpetas dentro de un directorio)`
- `ls -a (muestra los archivos o carpetas ocultas dentro de un directorio)`
- `ls -r (muestra los archivos o carpetas de un directorio de forma recursiva)`

Modificar (copiar, mover o renombrar):

- `mv [directorio o archivo o carpeta][directorio o archivo o carpeta] (renombra archivos o carpetas de no existir los mismos o, mueve archivos o carpetas a otra ubicación)`
- `cp [directorio o archivo o carpeta] [directorio o archivo o carpeta] (copia un archivo o carpeta dentro de la ubicación deseada)`

Con `-r`, copia o mueve recursivamente el contenido

En cada caso se debe especificar la ruta del archivo, carpeta o directorio que se quiera mover o copiar.

- cd .. (permite subir un nivel de carpeta)
- cd [nombre carpeta] (se utiliza para cambiar de carpeta)

Para nombras un archivo o carpeta cuyo nombre tenga más de una palabra, se encierra el mismo entre comillas (“nombre archivo”)

El punto hace referencia al lugar en que estamos ubicados

Borrar:

- rm [archivo o carpeta] (borra el archivo o carpeta que deseamos)
- rm -r[directorio] (borra el directorio y todas las carpetas y archivos que lo contienen)

Mostrar:

- git log (muestra todo el historial)
- git log [directorio] (muestra el historial de commits del directorio que deseamos ver)

- git checkout [archivo o carpeta] (deshace un cambio en un archivo o carpeta que se haya realizado en el repositorio local, y solo funciona si no se realizó el cambio en el repositorio remoto)
- git remote (muestra los repositorios remotos)
- git remote add origin <git@github.com:minombre/archivo-git.git> (enlaza el repositorio local con un repositorio remoto)
- git push -u origin master (el primer push en el repositorio debe contener su nombre y branch)
- git push (cuando se desea realizar un push del repositorio local al repositorio remoto)
- git pull (actualiza los archivos o carpetas del repositorio remoto en el repositorio local)
- git clone <git@github.com:minombre/archivo-git.git> (para clonar un repositorio remote ya existente)
- git branch nombreBranch (crea una nueva branch dentro de un repositorio)
- git checkout nombreBranch (permite realizar cambios en una branch ya existente)
- git checkout master (vuelve a la branch principal)
- git branch -d nuevaBranch (da de baja esa rama)
- git branch (lista ramas existentes dentro de un repositorio)
- git pull origin nombreBranch (actualiza los archivos o carpetas de la rama en el repositorio local)
- git push origin nuevaBranch (actualiza los archivos o carpetas desde el repositorio local a la branch)