

# CHEATSHEET RECORDATORIO DE COMANDOS DE GIT

## COMANDOS BÁSICOS DE LA TERMINAL GIT BASH

MOSTRAR
Lista archivos en el directorio o carpeta
\$ ls
Lista todos los archivos, incluyendo los archivos ocultos
\$ ls -a
Muestra toda la información de una carpeta: usuario, grupo, permisos, tamaño, fecha y hora de creación.
\$ ls -l
Muestra las carpetas y los archivos contenidos en ellos de manera recursiva
\$ ls -R
Muestra la carpeta en la que se está trabajando actualmente
\$ pwd
Muestra el contenido de un archivo
more [Nombre del archivo]

CREAR
Crea una nueva directorio o carpeta
\$ mkdir [Nombre_Carpeta]
Crea un nuevo archivo
\$ touch [Nombre_archivo]

ELIMINAR
Elimina un archivo
\$ rm [Nombre del archivo]
Elimina una carpeta vacía
\$ rmdir [Nombre de la carpeta]
Elimina una carpeta y su contenido
\$ rm -r [Nombre de la carpeta]

COPIAR / MOVER / RENOMBRAR
Renombra archivos (archivo2 no debe existir o será sobrescrito)
\$ mv [ruta/archivo1] [ruta/archivo2]
Renombra la carpeta1 como carpeta2 (carpeta 2 no debe existir)
\$ mv [ruta/carpeta1] [ruta/carpeta2]
Mueve contenido de carpeta1 a carpeta2 (carpeta 2 debe existir)
\$ mv [ruta/carpeta1] [ruta/carpeta2]
Copia un archivo o carpeta
\$ cp [ruta/archivo1] [ruta/archivo2]
\$ cp [ruta/archivo1] [ruta/archivo2]
Indica que copie recursivamente el contenido de las subcarpetas
opción: -r

NAVEGACIÓN ENTRE CARPETAS
Sube un nivel de carpeta
\$ cd ..
Cambia de carpeta
\$ cd [Nombre_de_Carpeta]
Cambia a una carpeta específica
\$ cd/chosen/directory

CARACTERES ESPECIALES
Limpia la pantalla de la terminal
\$ clear
Muestra ayuda del comando
\$ comando --help
Nos permiten utilizar términos que consistan en más de una palabra
"" (comillas)
Permite hacer referencia al directorio donde estamos ubicados actualmente
. (el punto)

# CHEATSHEET RECORDATORIO DE COMANDOS DE GIT

## COMANDOS PRINCIPALES PARA GIT

CREAR / CLONAR UN REPOSITORIO
Clonar un repositorio remoto
\$ git clone [URL del repositorio remoto]
Inicializar un repositorio
\$ git init

CONFIGURAR USUARIO Y EMAIL
Definir usuario
\$ git config user.name "nombreDeUsuario"
Verificar si se agregó nuevo usuario
\$ git config user.name
Definir email (mismo de cuenta GitHub)
\$ git config user.email "nombre@email.com"
Verificar si se agregó nuevo correo
\$ git config user.email
Configurar usuario global
\$ git config --global user.name "nombreDeUsuario"
Configurar email global
\$ git config --global user.email "nombre@email.com"

ESTADO DE ARCHIVOS, AÑADIR CAMBIOS Y HACER COMMIT
Ver estado de archivos de repositorio
\$ git status
Agregar cambios de un archivo específico
\$ git add [Nombre_del_Archivo.extension]
Agregar cambios de todos los archivos
\$ git add .
Confirmar los cambios (Commit)
\$ git commit -m "mensaje"

CONECTAR REPOSITORIO LOCAL CON REPOSITORIO REMOTO
Conectar repositorios
\$ git remote add origin [URL_Repositorio_Remoto]
Verificar que estan conectados
\$ git remote -v

BAJAR Y SUBIR CAMBIOS A REPOSITORIO REMOTO
Traer archivos de repositorio remoto
\$ git pull origin main
Enviar archivos a repositorio remoto
\$ git push origin main

TRABAJANDO CON RAMAS
Crear una nueva rama
\$ git branch [Nombre_Rama]
Enumera todas las ramas de tu repositorio
\$ git branch
Eliminar una rama específica
\$ git branch -d [Nombre_Rama]
Forzar la eliminación de una rama
\$ git branch -D [Nombre_Rama]
Moverse de una rama a otra
\$ git checkout [Nombre_Rama]
Traer archivos de una rama de repositorio remoto
\$ git pull origin [Nombre_Rama]
Enviar archivos de una rama a repositorio remoto
\$ git push origin [Nombre_Rama]