

# CHEAT SHEET



COMANDO	DESCRIPCIÓN	SINTAXIS
git config	Configura el nombre de usuario en el proyecto	\$ git config user.name "nombre_de_usuario"
	Para verificar nombre de usuario en el proyecto	\$ git config user.name
	Configura el email del usuario en el proyecto	\$ git config user.email "nombre@email.com"
	Para verificar email de usuario en el proyecto	\$ git config user.email
git config --global	Configura el nombre de usuario de manera global	\$ git config --gobal user.name "nombre_de_usuario"
	Configura el email de usuario de manera global	\$ git config --gobal user.email "nombre@email.com"
git config --list	Muestra la lista de configuración de git	\$ git config --list
	Rutas de acceso a la configuración de git	\$ git config --list --show-origin
git remote	Conectar repositorio local con repositorio remoto	\$ git remote add origin
	Verificar conexión repositorio local con repositorio remoto	\$ git remote -v
git clone	Para clonar un repositorio remoto y descargarlo por primera vez a nuestra máquina	\$ git clone ssh://usuario@domain.com/repo.git
git init	Inicializa un nuevo repositorio	\$ git init
git add	Agrega los cambios de un archivo al repositorio	\$ git add nombre_de_archivo.extensión
	Agrega todos los archivos al repositorio	\$ git add .
git commit	Agrega los cambios al repositorio	\$ git commit -m "Mensaje"
git status	Para ver el estado de nuestros archivos	\$ git status
git log	Histórico de cambios con detalles	\$ git log nombre_de_archivos.extensión
git pull	Actualiza los archivos de nuestro repositorio local con respecto a los que los están en el repositorio remoto	\$ git pull
	Actualiza los archivos de una rama especifica de nuestro repositorio local con respecto a lo que está en el repositorio remoto	\$ git pull origin -nombre_de_la_rama-
git push	están en el repositorio remoto	\$ git push
	Envía los archivos de una rama especifica al repositorio remoto	\$ git push origin -nombre_de_la_rama-
git checkout	Trae cambios realizados	\$ git checkout
git rm --cached	Se utiliza para devolver el archivo que se tiene en ram. Cuando escribimos git add, lo devuelve a estado natural mientras está en staging.	\$ git rm --cached archivo.extensión
git log	Muestra la historia del archivo	\$ git log archivo.extensión
git branch	Genera una nueva rama	\$ git branch -nombre de la rama-
git checkout	Salta de una rama a otra	\$ git checkout -nombre de la rama-
git checkout -b	Genera una rama y nos mueve a ella automáticamente, Es decir, es la combinación de git brach y git checkout al mismo tiempo.	\$ git checkout -b rama