



## CHEAT SHEET GIT

Git y Github: Control de versiones de la A a la Z

### Git y Github: Control de versiones de la A a la Z

Git es el sistema de control de versiones distribuido, gratuito y de código abierto, responsable de todo lo relacionado con GitHub que ocurre localmente en el ordenador. Esta hoja de trucos presenta los comandos de Git más importantes y más utilizados para una fácil referencia.

### CONFIGURACIÓN

<code>git config --global user.name "[nombre-apellido]"</code>	Establecer un nombre que sea identificable para cuando se revise el historial de versiones
<code>git config --global user.email "[email válido]"</code>	Establecer una dirección de correo electrónico que se asociará a cada commit del historial
<code>git config --global -e</code>	Fichero con la información de configuración de Git

### INICIALIZACIÓN

<code>git init</code>	Inicializar un directorio existente como repositorio Git
<code>git clone [url]</code>	Recuperar un repositorio completo desde una ubicación alojada a través de una URL

### ESCENARIO

<code>git status</code>	Mostrar los archivos modificados en el directorio de trabajo, preparados para su el próximo commit
<code>git add [fichero]</code>	Añadir un archivo para el próximo commit (escenario)
<code>git add -u</code>	Actualiza el índice sólo donde ya tiene una entrada que coincide con <pathpec>.
<code>git reset [fichero]</code>	Desenvolver un archivo conservando los cambios en el directorio de trabajo
<code>git reset -- soft HEAD^</code>	Desenvolver el último commit conservando los cambios en el directorio local
<code>git diff</code>	Diferencia de lo que se cambia pero no se pone en escena
<code>git diff --staged</code>	Diferencia de lo que se ha puesto en escena pero aún no se ha hecho commit
<code>git commit -m "[mensaje descriptivo]"</code>	Hacer un nuevo commit con su respecta descripción

### RAMAS

<code>git branch</code>	Listar las ramas y aparecerá un * al lado de la rama actualmente activa
<code>git branch [nombre de la rama]</code>	Crear una nueva rama en el commit actual
<code>git checkout -b [nombre de la rama]</code>	Creación y desplazamiento a la rama indicada
<code>git branch -d [nombre de la rama]</code>	Elimina la rama indicada
<code>git checkout</code>	Cambiar a otra rama y comprobarla en el directorio de trabajo
<code>git checkout -- .</code>	Retomar el estado del último commit
<code>git merge [rama]</code>	Fusionar el historial de la rama especificada en la actual
<code>git log</code>	Mostrar todos los commits en el historial de la rama actual



## CHEAT SHEET GIT

Git y Github: Control de versiones de la A a la Z

### INSPECCIONAR Y COMPARAR

<code>git log</code>	Mostrar el historial de commits de la rama actualmente activa
<code>git log ramaB...ramaA</code>	Mostrar los commits de la ramaA que no están en la rama B
<code>git log --follow [file]</code>	Mostrar los commits que cambiaron el archivo, incluso a través de los renombramientos
<code>git diff ramaB...ramaA</code>	Mostrar la diferencia de lo que hay en la rama A que no está en la rama B
<code>git show [SHA]</code>	Mostrar cualquier objeto en Git en formato legible para el ser humano

### SEGUIMIENTO DE LOS CAMBIOS

<code>git rm [fichero]</code>	Eliminar el archivo del proyecto y preparar la eliminación para el commit
<code>git mv [ruta existente] [nueva ruta]</code>	Renombrar un archivo o directorio o cambiar la ruta de un archivo existente y realizar el movimiento
<code>git log --stat -M</code>	Mostrar todos los registros de commit con indicación de las rutas que se han movido

### IGNORANDO LOS PATRONES

<code>logs/</code> <code>*.notes</code> <code>pattern*/</code>	Guarda un archivo con los patrones deseados como .gitignore
<code>git config --global core.excludesfile [fichero]</code>	Git ignora ese fichero para todos los repositorios

### COMPARTIR Y ACTUALIZAR

<code>git remote add [alias] [url]</code>	Añadir una URL de git como alias
<code>git fetch [alias]</code>	Obtener todas las ramas de ese Git remoto
<code>git merge [alias]/[rama]</code>	Fusionar una rama remota en su rama actual para actualizarla
<code>git push [alias] [rama]</code>	Transmitir los commits de la rama local a la rama del repositorio remoto
<code>git pull</code>	Obtener y fusionar los commits de la remota a la rama local

### TAGS

<code>git tag</code>	Visualizar todos los tags
<code>git tag -a [ID del tag] -m "[mensaje descriptivo]"</code>	Crear un tag para el último commit realizado con un mensaje enlazado a este tag
<code>git tag -a [ID del tag] [hash del commit] -m "[mensaje descriptivo]"</code>	Crear un tag para un commit identificado con su hash
<code>git tag -d [ID del tag]</code>	Eliminar el tag especificado
<code>git push --tags</code>	Enviar todos los tags al repositorio remoto

### REESCRIBIR LA HISTORIA

<code>git rebase [rama]</code>	Aplicar cualquier commit de la rama actual por delante de la especificada
<code>git reset --hard [commit]</code>	Despejar el escenario y reescribir el árbol de trabajo a partir del commit especificado



## CHEAT SHEET GIT

Git y Github: Control de versiones de la A a la Z

### COMMITTS TEMPORALES

git stash	Almacenamiento de los cambios realizados en la "pila de almacenamiento", para centrarse en otro campo y volver más adelante al punto anterior
git stash list	Listar el orden de pila de los cambios en los archivos almacenados
git stash pop	Toma un cambio almacenado, lo elimina de la "pila de almacenamiento" y lo aplica al árbol de trabajo actual
git stash drop	Descartar los cambios de la parte superior de la "pila de almacenamiento"
git stash clear	Borrar todas las entradas del stash list
git stash apply	Restaurar el último registro del stash
git stash apply [ID del stash]	Restaurar el stash identificado con su ID (por ejemplo, ID: stash@{0})
git stash save --keep-index	Guardar todos los archivos menos los archivos del stage
git stash save --include-untracked	Incluir todos los archivos, incluso a los que git no da seguimiento
git stash list --stat	Aparece más información para cada uno de los stash de los registros
git stash branch [nombre de la rama] [ID-Stash]	Creación de una rama a partir de un stash identificado con su ID
git stash show	Información detallada del stash