

Comandos de Git

Git es un sistema de control de versiones distribuido, gratuito y de código abierto, creado por Linus Torvalds en 2005, diseñado para el manejo de todo tipo de proyectos, desde proyectos pequeños a muy grandes, con velocidad y eficiencia. La presente hoja de referencia resume los comandos de Git más comúnmente utilizados:

Comando	Significado
Configuración de Git	
git config --global user.name "username"	Establece el nombre de usuario de forma global para todos los repositorios locales.
git config --global user.name "example@email.com"	Establece el correo electrónico del usuario de forma global para todos los repositorios locales.
Crear o clonar un repositorio	
git init [project name]	Inicializa un repositorio local
git clone repository_url	Clonar un repositorio existente
Observar cambios en el repositorio	
git status	Muestra el estado de los archivos en el repositorio, es decir, la lista de archivos nuevos o modificados, que aun no se han agregado o commiteado.
git diff	Mostrar los cambios en los archivos que aún no se han enviado al area de espera, es decir, que aún no están en el staging area.
git diff commit1 commit2	Muestra los cambios o diferencias entre los ID's de dos commits (commit1 y commit2).
git show [commit]:[file]	Mostrar los cambios de un archivo para un commit o archivo específico.
git log	Muestra el historial de cambios completo.
Agregar cambios en el repositorio	
git add [file]	Agrega un archivo al staged area, dejándolo listo para ser commiteado.

git add .	Agrega todos los archivos modificados, dejándolos listos para ser commiteados. .
git commit -m "commit message"	Registra los cambios en el repositorio, tomando una instantánea de como se encuentran los archivos en ese momento. Este comando guardará todos los cambios que se encuentran en el staging area.
Sincronizar cambios	
git fetch	Obtiene los últimos cambios desde el repositorio remoto (sin hacer un merge).
git pull	Descarga los cambios que existen en el repositorio remoto. Con este comando, se obtienen los últimos cambios desde el repositorio remoto y se realiza un merge.
git push origin branch_name	Envía los cambios desde el repositorio local al repositorio remoto.
Trabajando con ramas	
git branch	Lista todas las ramas locales
git checkout my_branch	Este comando sirve para cambiar de una rama a otra y actualizar el directorio de trabajo. En este caso, hace el cambio a la rama con el nombre my_branch.
git branch new_branch	Crea una nueva rama con el nombre new_branch
git branch -d my_branch	Elimina la rama llamada my_branch
git checkout branch_b git merge branch_a	Fusiona la rama branch_a con la rama branch_b