Guía Completa de Comandos de Git

1. Configuración Inicial (Se hace una sola vez)

Estos comandos configuran tu identidad en Git para todos tus proyectos.

- **`git config --global user.name "Tu Nombre"`**
  - Establece el nombre que se mostrará como autor de tus commits.
- **`git config --global user.email "tu_correo@ejemplo.com"`**
  - Establece el correo electrónico que se asociará a tus commits.
- **`git config --global init.defaultBranch main`**
  - Configura `main` como el nombre por defecto para la rama principal de nuevos repositorios.
- **`git config --global fetch.prune true`**
  - (Opcional) Hace que al actualizar (`fetch`), se limpien automáticamente las ramas que ya no existen en el remoto.
- **`git config --list`**
  - Muestra tu configuración actual para verificarla.

2. Crear y Conectar un Repositorio a GitHub

Sigue estos pasos para un proyecto nuevo que quieres subir a un repositorio recién creado en GitHub.

- **`git init`**
  - Inicializa un repositorio de Git vacío en tu carpeta actual.
- **`git add .`** o **`git add README.md`**
  - Agrega los archivos al área de preparación (staging). Usar `.` agrega todos los archivos del directorio.
- **`git commit -m "first commit"`**
  - Guarda los archivos preparados en el historial de tu repositorio local con un mensaje descriptivo.
- **`git branch -M main`**
  - (Opcional pero recomendado) Renombra la rama actual a `main`.
- **`git remote add origin [URL_del_repositorio.git]`**
  - Conecta tu repositorio local con el repositorio remoto en GitHub. Reemplaza la URL con la de tu propio repositorio.
- **`git push -u origin main`**
  - Sube por primera vez tus commits a la rama `main` del repositorio remoto (`origin`). La opción `-u` establece la conexión para que en el futuro solo necesites usar `git push`.

3. Flujo de Trabajo Básico (Los más usados en el día a día)

- **git clone [URL_del_repositorio.git]**
  - Crea una copia local de un repositorio remoto existente.
- **git status**
  - Muestra el estado de tus archivos (modificados, en preparación, sin seguimiento).
- **git add [archivo]** o **git add .**
  - Agrega los cambios de uno o todos los archivos al área de preparación para el próximo commit.
- **git commit -m "Mensaje descriptivo"**
  - Guarda los cambios preparados en el historial local.

4. Trabajo con Ramas (Branches)

- **git branch**
  - Lista todas las ramas locales. La actual se marca con un *.
- **git branch [nombre-nueva-rama]**
  - Crea una nueva rama.
- **git checkout [nombre-rama]**
  - Cambia a la rama especificada.
- **git checkout -b [nombre-nueva-rama]**
  - Crea una nueva rama y cambia a ella en un solo paso.
- **git merge [nombre-rama-a-fusionar]**
  - Incorpora los cambios de otra rama en la rama actual.
- **git branch -d [nombre-rama]**
  - Elimina una rama local (solo si ya ha sido fusionada).

5. Sincronización con Repositorios Remotos

- **git pull**
  - Descarga los cambios del repositorio remoto y los fusiona con tu rama local actual.
- **git push**
  - Sube tus commits locales al repositorio remoto.
- **git fetch**
  - Descarga la información del remoto (nuevas ramas y commits) pero no la fusiona con tu trabajo local.
- **git fetch --prune**
  - Actualiza la información del remoto y elimina las referencias a ramas que ya fueron borradas en el servidor.
- **git remote -v**
  - Muestra las URLs de los repositorios remotos conectados.
- **git push origin --delete [nombre-rama]**
  - Elimina una rama en el repositorio remoto.

6. Historial y Reversión de Cambios

- **git log**
  - Muestra el historial de commits de la rama actual.
- **git log --oneline**
  - Muestra el historial de forma compacta, con un commit por línea.
- **git log --graph --oneline --all**
  - Muestra el historial de todas las ramas de forma gráfica y compacta.
- **git reset --hard [hash_del_commit]**
  - **(Comando peligroso)** Revierte el proyecto a un commit anterior, descartando permanentemente todos los cambios locales que no estén guardados en un commit.
- **git reset --hard HEAD**
  - Descarta todos los cambios locales y vuelve al estado del último commit.