## 1️⃣ Ver el estado del repositorio

git status

Muestra los cambios **pendientes**, archivos sin seguimiento y en qué rama estás.

Útil para saber qué se ha modificado antes de hacer commit.

## 2️⃣ Preparar los cambios para commit

git add .

Prepara **todos los archivos nuevos y modificados** para ser confirmados.

También puedes usar git add <archivo> para agregar solo un archivo específico.

## 3️⃣ Guardar cambios en un commit

git commit -m "Mensaje descriptivo"

Guarda tus cambios localmente en Git con un **mensaje que describa lo que hiciste**.

Ejemplo:

git commit -m "Primer commit: entorno de desarrollo listo"

## 4️⃣ Crear ramas

git checkout -b nombre-rama

Crea una **nueva rama** y te mueve automáticamente a ella.

Se usa para trabajar en funcionalidades nuevas sin afectar la rama principal (master o main).

Ejemplo:

git checkout -b rama-Keren

## 5️⃣ Cambiar de rama

git checkout nombre-rama

Te mueve a otra rama existente.

Siempre asegúrate de estar en la rama correcta antes de hacer cambios o commits.

## 6️⃣ Eliminar una rama local

git branch -D nombre-rama

Borra una rama local **incluso si no tiene commits fusionados**.

No puedes borrar la rama en la que estás actualmente.

## 7️⃣ Conectar tu repositorio local con GitHub

git remote add origin https://github.com/TU-USUARIO/TU-REPO.git

Esto vincula tu proyecto local con el repositorio remoto en GitHub.

Solo se hace **una vez por proyecto**.

git remote -v

Verifica que el remoto (origin) se haya agregado correctamente.

## 8️⃣ Subir ramas a GitHub

git push -u origin nombre-rama

Sube tu rama local al repositorio remoto en GitHub.

-u vincula la rama local con la remota, para que luego solo necesites git push.

Ejemplo:

git push -u origin rama-Keren

## 9️⃣ Ver todas las ramas (locales y remotas)

git branch -a

Lista todas las ramas **locales** y **remotas**.

Las ramas remotas aparecen como remotes/origin/nombre-rama.

git ls-remote --heads origin

Lista únicamente las ramas que existen **en el repositorio remoto** (GitHub).

## 10️⃣ Recuperar un archivo borrado accidentalmente

git restore nombre-archivo

Revierte cambios o restaurar archivos eliminados que Git está rastreando.

## 11️⃣ Descartar cambios no deseados

git restore nombre-archivo

Deshace los cambios locales de un archivo específico.

### ✅ Flujo general que seguimos

git status → revisar cambios y ramas.

git add . → preparar archivos para commit.

git commit -m "mensaje" → guardar cambios localmente.

git checkout -b rama-nueva → crear y moverse a una rama.

git checkout master → volver a la rama principal para crear otras ramas.

git branch -D rama-equivocada → eliminar ramas incorrectas.

git remote add origin URL → vincular con GitHub.

git push -u origin rama-nueva → subir la rama al repositorio remoto.

git branch -a → verificar todas las ramas locales y remotas.