

Git Remote

Para agregar un repositorio remoto:

```
git remote add origin <url>
```

Para cambiar de remote:

```
git remote set-url origin <url>
```

Para remover repositorio:

```
git remote rm <name/origin>
```

Git Clone

Descarga el código fuente existente desde un repositorio remoto. Osea, realiza una copia idéntica de la última versión de un proyecto en un repositorio y la guarda en el ordenador.

```
git clone <https://link-con-nombre-del-repositorio>
```

Git branch

Crear una nueva rama local:

```
git branch <nombre-de-la-rama>
```

Para enviar la nueva rama al repositorio remoto:

```
git push <nombre-remoto> <nombre-rama>
```

Visualización de ramas:

```
git branch
```

Borrar una rama:

```
git branch -d <nombre-de-la-rama>
```

Git checkout

Para cambiar de rama. Antes de hacerlo, se deben confirmar o almacenar en el guardado rápido (stash) los cambios de la rama actual.

```
git checkout <nombre-de-la-rama>
```

Otro comando para crear y cambiarse a esa rama al mismo tiempo:

```
git checkout -b <nombre-de-tu-rama>
```

Git status

Muestra toda la información necesaria sobre la rama actual

- Si la rama actual está actualizada
- Si hay algo para confirmar, enviar o recibir (pull).
- Si hay archivos en preparación (staged), sin preparación(unstaged) o que no están recibiendo seguimiento (untracked)
- Si hay archivos creados, modificados o eliminados

```
git status
```

Git add

Para incluir los cambios realizados en el siguiente commit.

Añadir un único archivo:

```
git add <archivo>
```

Añadir todos a la vez:

```
git add .
```

Git commit

Para guardar definitivamente los cambios en un commit (únicamente en local). Se establece un punto de control en el proceso de desarrollo al cual se puede volver más tarde si es necesario.

```
git commit -m "mensaje de confirmación"
```

Para ver los gráficos de los commits:

```
git log --oneline --graph
```

Saca un archivo del commit:

```
git reset HEAD <archivo>
```

Git push

Para enviar los cambios (commits) al repositorio remoto. Solamente carga los cambios que han sido confirmados.

```
git push <nombre-remoto> <nombre-de-tu-rama>
```

Git pull

Para recibir actualizaciones del repositorio remoto.

```
git pull <nombre-remoto>
```

Git revert

Para ver el historial de commits:

```
git log--online
```

```
Cem-MacBook-Pro:my-new-app cem$ git log --oneline
3321844 (HEAD -> master) test
e64e7bb Initial commit from Create React App
```

Se deberá especificar el código de comprobación que se encuentra junto al commit que se quiera deshacer:

```
git revert 3321844
```

```
Revert "test"

This reverts commit 332184490ef2b5db289d85ed3f1a13ff2d5f94b9.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Committer: Cem <cem@Cem-MacBook-Pro.local>
#
# On branch master
# Changes to be committed:
#   modified:   src/App.js
#   deleted:    src/components/myFirstComponent.js
#
~
~
~
~
~
~
~
~/Desktop/my-new-app/.git/COMMIT_EDITMSG" 14L, 384C
```

Git merge

Para fusionar ramas. Integra las características de la rama con todos los commits realizados a las ramas main. Para fusionar con la main se debe estar posicionado en ella y especificar la rama a combinar.

IMPORTANTE: actualizar la rama main a la última versión con **git fetch**. Sino, ocurrirán conflictos no deseados.

```
git merge <nombre-de-la-rama>
```

