

git clone

Este comando sirve para obtener una copia del *fork* que has hecho previamente en tu máquina local.

```
● @lucia-alvarez-castro → /workspaces/p1 (main) $ git clone https://github.com/gitt-3-pat/p1
Cloning into 'p1'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 5
Receiving objects: 100% (6/6), done.
○ @lucia-alvarez-castro → /workspaces/p1 (main) $ █
```

git status

Este comando sirve para obtener información sobre el estado actual del repositorio.

```
● @lucia-alvarez-castro → /workspaces/p1-fork (main) $ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit. working tree clean
```

En este caso, la salida del comando `git status` indica que mi repositorio está actualizado, que no hay cambios pendientes y que todo está limpio.

git add, git commit y git push

Los cambios en cada rama de un repositorio se llevan a cabo en tres pasos: `add`, `commit`, y `push`. En este caso, he creado un directorio llamado `src` con un nuevo archivo `html` dentro. Después, con `add`, `commit` y `push` lo he clonado al repositorio remoto.

```
● @lucia-alvarez-castro → /workspaces/p1/src (main) $ git add .
● @lucia-alvarez-castro → /workspaces/p1/src (main) $ git commit -m "feat: homepage"
[main 26e27e2] feat: homepage
2 files changed, 6 insertions(+)
create mode 160000 p1
create mode 100644 src/index.html
● @lucia-alvarez-castro → /workspaces/p1/src (main) $ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 424 bytes | 424.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/lucia-alvarez-castro/p1
07720b5..26e27e2 main -> main
```

En este caso, mi repositorio remoto es:

```
● @lucia-alvarez-castro → /workspaces/p1-fork/src (main) $ git remote -v
origin https://github.com/lucia-alvarez-castro/ci-cd.git (fetch)
origin https://github.com/lucia-alvarez-castro/ci-cd.git (push)
```

git checkout

Con este comando se puede crear y acceder a una nueva rama. En este caso, se ha creado y se ha cambiado a la rama `feature/1`, y a continuación se ha vuelto a cambiar a la rama `main`.

```
● @lucia-alvarez-castro → /workspaces/p1-fork (main) $ git checkout -b feature/1
Switched to a new branch 'feature/1'
○ @lucia-alvarez-castro → /workspaces/p1-fork (feature/1) $

● @lucia-alvarez-castro → /workspaces/p1-fork (feature/1) $ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```