

## MEMORIA COMANDOS PRÁCTICA 1

A lo largo de esta memoria explicaremos brevemente los comandos utilizados en la práctica e incluiremos una captura del código introducido en la terminal.

### 1.Git Clone

```
● @paofcheca →/workspaces/ci-cd (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 (from 1)
Receiving objects: 100% (6/6), done.
○ @paofcheca →/workspaces/ci-cd (main) $
```

Este comando nos permite clonar un repositorio a partir de su URL en el nuestro.

### 2.Git Status

```
● @paofcheca →/workspaces/ci-cd (main) $ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  p1/

nothing added to commit but untracked files present (use "git add" to track)
○ @paofcheca →/workspaces/ci-cd (main) $
```

Git Status muestra el estado actual del repositorio local. Incluye cambios que aún no se han guardado (staged o unstaged), la rama en la que estás trabajando o archivos nuevos que no están siendo rastreados.

### 3.Git Add .

```
● @paofcheca →/workspaces/ci-cd (main) $ git add .
warning: adding embedded git repository: p1
hint: You've added another git repository inside your current repository.
hint: Clones of the outer repository will not contain the contents of
hint: the embedded repository and will not know how to obtain it.
hint: If you meant to add a submodule, use:
hint:
hint:   git submodule add <url> p1
hint:
hint: If you added this path by mistake, you can remove it from the
hint: index with:
hint:
hint:   git rm --cached p1
hint:
hint: See "git help submodule" for more information.
hint: Disable this message with "git config advice.addEmbeddedRepo false"
```

Git Add añade archivos al área de preparación (staging area). Esto prepara a los archivos para ser incluidos en el próximo commit.

#### 4. Git Commit

```
• @paofcheca →/workspaces/ci-cd (main) $ git commit
[main 9c8309e] Aquí escribo un mensaje de confirmación
1 file changed, 1 insertion(+)
create mode 160000 p1
```

Este comando guarda los cambios en el historial del repositorio. Un commit representa un punto de control del desarrollo.

#### 5. Git Push

```
• @paofcheca →/workspaces/ci-cd (main) $ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 355 bytes | 355.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/paofcheca/ci-cd
0733ea2..6779521 main -> main
```

Git Push sincroniza los cambios y envía los commits locales al repositorio remoto.

#### 6. Git Checkout

```
• @paofcheca →/workspaces/ci-cd (main) $ git checkout
Your branch is up to date with 'origin/main'.
```