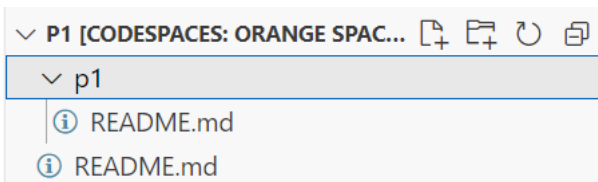


## Comandos importantes

### - Comando git clone (url repositorio github)

El comando "git clone https://github.com/gitt-3-pat/p1" lo utilizo para copiar un repositorio remoto, en concreto el enunciado de la práctica 1, a mi máquina local. El resultado que he obtenido tras ejecutarlo en mi terminal indica que el clonado fue exitoso, y se ha creado un directorio llamado 'p1' en mi repositorio local. En los logs que ha dejado en la consola tras su ejecución podemos ver "transfiriendo 6 objetos" lo cual indica lo que ha clonado desde el repositorio remoto al local. Indican también que no hubo cambios (delta 0) y se optimizó la transferencia (pack-reused 5). A continuación, podemos ver una imagen del resultado obtenido:

```
@ajm463 → /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.
```



### - Comando git status

El resultado de git status indica que estoy en la rama main y que mi rama está actualizada con la rama origin/main en el repositorio remoto, que en este caso al haber hecho fork están en mi propia cuenta. Sin embargo, también muestra que hay archivos no rastreados en tu directorio de trabajo. Esto se soluciona con el siguiente comando.

```
@ajm463 → /workspaces/p1 (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)
```

### - Comando git add .

Al ejecutar este comando estoy añadiendo todos los cambios en mi directorio de trabajo. El punto indica que voy a agregar todos los cambios en el directorio actual al área de preparación.

```
@ajm463 →/workspaces/p1 (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.
```

(Vemos el cambio en status con git status una vez añadido p1/

```
@ajm463 →/workspaces/p1 (main) $ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   p1
```

## - Comando git commit -m "HOLA"

He realizado un commit en la rama main con el mensaje "HOLA". Este commit ha al archivo llamado p1 una línea con el mensaje "HOLA" en ese archivo. Si utilizo git push para enviar este commit a tu repositorio remoto en GitHub puedo mantener sincronizados tus cambios locales y remotos.

```
@ajm463 →/workspaces/p1 (main) $ git commit -m "HOLA"
[main 4c87cc2] HOLA
1 file changed, 1 insertion(+)
create mode 160000 p1
```

```
@ajm463 →/workspaces/p1 (main) $ git show 4c87cc2
commit 4c87cc2f58278505ace4eb17967586dac6c8b93f (HEAD -> main)
Author: ajm463 <157055993+ajm463@users.noreply.github.com>
Date:   Fri Jan 26 16:48:44 2024 +0000

    HOLA

diff --git a/p1 b/p1
new file mode 160000
index 0000000..07720b5
--- /dev/null
+++ b/p1
@@ -0,0 +1 @@
+Subproject commit 07720b55268c921986ba40aaf84ff160295d0bc5
```

```
@ajm463 → /workspaces/p1 (main) $ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

Vemos que el repositorio actual difieren por un commit realizado en este comando.

## - Comando git push

Con este comando aplicamos los cambios al repositorio remoto, el cual al haber hecho fork está en mi cuenta por lo que aparecerán ahí, tal como podemos ver en la segunda captura a continuación

```
@ajm463 → /workspaces/p1 (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), 275 bytes | 275.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/ajm463/p1
  07720b5..4c87cc2  main -> main
```

The screenshot shows a GitHub repository page for user 'ajm463' with the repository name 'HOLA'. The commit hash is '4c87cc2' and it was made '6 minutes ago'. The commit message is 'HOLA'. Below the commit message, there is a list of files changed: 'p1' (added) and 'README.md' (updated). The 'p1' file is highlighted, showing its content: 'HOLA'. The commit is linked to the 'main' branch. Below the file list, there is a section for 'Showing 1 changed file with 1 addition and 0 deletions'. The file 'p1' is shown with its content: 'Submodule p1 added at 07720b'. At the bottom, there is a comment section with a 'Write' tab and a 'Preview' tab. The 'Write' tab is active, showing a text area for leaving a comment. A green button 'Comment on this commit' is visible at the bottom right.

## - Comando git checkout -b feature/1

El comando git checkout -b feature/1 se utiliza para crear una nueva rama llamada feature/1 y cambiar a ella en un solo paso.

```
@ajm463 → /workspaces/p1 (main) $ git checkout -b feature/1
Switched to a new branch 'feature/1'
```

## - **Comando git checkout main**

El comando git checkout main se utiliza para cambiar de rama a la rama llamada main. Al ejecutar este comando, te moverás de la rama actual a la rama main.

```
@ajm463 → /workspaces/p1 (feature/1) $ git checkout main  
Switched to branch 'main'  
Your branch is up to date with 'origin/main'.
```