

Práctica 1

```
• @miriamicai →/workspaces/p1-fork (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.
• @miriamicai →/workspaces/p1-fork (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/

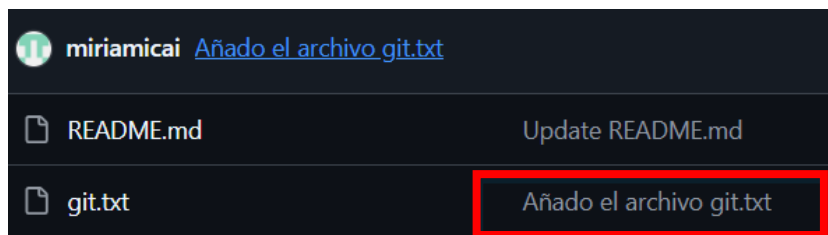
nothing added to commit but untracked files present (use "git add" to track)
```

El comando **git clone** sirve para crear una copia local de un repositorio. En este caso para realizar la prueba, se ha clonado dentro del repositorio p1-fork el repositorio original p1 por lo que se crea un subdirectorio, que es a su vez un repositorio de Git.

El comando **git status** sirve para ver el estado actual de la rama en la que se está ejecutando el comando, pudiendo ver si está sincronizada con el repositorio remoto.

```
• @miriamicai →/workspaces/p1-fork (main) $ git add .
• @miriamicai →/workspaces/p1-fork (main) $ git commit -m "Añado el archivo git.txt"
[main 9bea7cf] Añado el archivo git.txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 git.txt
• @miriamicai →/workspaces/p1-fork (main) $ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 293 bytes | 293.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/miriamicai/p1-fork
 07720b5..9bea7cf  main -> main
```

El comando **git add .** sirve para seleccionar todos los cambios que se han hecho para posteriormente ejecutar **git commit -m "mensaje"** que sirve para confirmar los cambios que se han hecho en el repositorio local. En el mensaje, se puede añadir una descripción de los cambios que se han hecho.



El comando **git push** sirve para enviar el commit realizado del repositorio local al remoto y de esta forma sincronizar ambos repositorios.

```
• @miriamicai →/workspaces/p1-fork (main) $ git branch nuevarama
• @miriamicai →/workspaces/p1-fork (main) $ git checkout nuevarama
  Switched to branch 'nuevarama'
• @miriamicai →/workspaces/p1-fork (nuevarama) $ git push -u origin nuevarama
  Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
  remote:
  remote: Create a pull request for 'nuevarama' on GitHub by visiting:
  remote:   https://github.com/miriamicai/p1-fork/pull/new/nuevarama
  remote:
  To https://github.com/miriamicai/p1-fork
   * [new branch]      nuevarama -> nuevarama
  branch 'nuevarama' set up to track 'origin/nuevarama'.
• @miriamicai →/workspaces/p1-fork (nuevarama) $ git checkout main
  Switched to branch 'main'
  Your branch is up to date with 'origin/main'.
```

Por último, el comando **git checkout** sirve para cambiar de una rama de trabajo a otra. También se puede combinar en **git checkout -b nuevarama** que crearía una nueva rama a la vez que cambiaría el directorio de trabajo a “nuevarama”.