

# Prueba de comandos de Git

**git clone:** Este comando clona mi fork.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas> git clone https://github.com/paulacollado/p1.git
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.
```

**git status:** Este comando muestra el estado actual del repositorio.

En primer lugar, usé este comando sin añadir ningún archivo y el resultado que obtuve fue el siguiente.

```
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas> cd p1
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
```

Como no había ningún cambio en el repositorio, el resultado del comando me indicaba que no había ningún archivo pendiente de ser añadido ni que hubiera ninguna modificación realizada, es decir, el repositorio estaba limpio.

Por ello, para realizar más pruebas con el comando, añadí manualmente un archivo llamado *Prueba.txt* en la carpeta del repositorio y ejecuté el comando de nuevo, obteniendo lo siguiente.

```
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> 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)
  Prueba.txt
```

El resultado mostraba que había un archivo “untracked” llamado *Prueba.txt*, que no formaba parte aún del repositorio ya que era necesario incluirlo mediando el comando “git add”.

**git add:** Este comando añade los archivos y los prepara para el commit.

Probé el comando añadiendo el archivo *Prueba.txt*, y al no devolver nada, indicó que se ejecutó correctamente la acción de añadir.

```
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> git add prueba.txt
```

**git commit:** Este comando registra los cambios en el repositorio con un mensaje.

El resultado de haber aplicado el comando fue el siguiente, que indicaba que funcionó correctamente en mi repositorio local, pero el cambio aún no estaba sincronizado con el repositorio remoto.

```
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> git commit -m "Añadido archivo Prueba.txt con explicación de comandos"
>>
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
```

**git push:** Este comando envía los cambios al repositorio remoto (GitHub).

```
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> git push
>>
info: please complete authentication in your browser...
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 316 bytes | 316.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/paulacollado/p1.git
  07720b5..4327dc1  main -> main
```

Este mensaje fue el que recibí tras realizar el “git push”, que me indicaba que los cambios en mi repositorio local se sincronizaron correctamente con el repositorio remoto en GitHub.

Antes de probar el comando “git checkout”, decidí realizar una prueba adicional eliminando el archivo *Prueba.txt* de la carpeta del repositorio y añadiendo un nuevo archivo llamado *git.txt*. Luego ejecuté nuevamente los comandos anteriores para reflejar estos cambios en el repositorio.

Antes de confirmar los cambios con “git add” y “git commit”, ejecuté “git status” para verificar el estado del repositorio, obteniendo lo siguiente.

```
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    Prueba.txt

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

Finalmente, confirmé los cambios en el repositorio local con los comandos “git add” y “git commit”, y los sincronice con el repositorio remoto mediante “git push”.

```
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> git add .
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> git commit -m "Añadido archivo git.txt y borrado el Prueba"
[main cb1c3b5] Añadido archivo git.txt y borrado el Prueba
1 file changed, 0 insertions(+), 0 deletions(-)
rename Prueba.txt => git.txt (100%)
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 298 bytes | 298.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/paulacollado/p1.git
4327dc1..cb1c3b5 main -> main
```

**git checkout:** Este comando permite cambiar entre ramas o recupera una versión anterior de un archivo.

Para probar el comando, realicé el comando siguiente; de manera que creé y cambié a una nueva rama llamada *nueva-rama*, obteniendo un mensaje de que se confirmaba esta acción.

```
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> git checkout -b nueva-rama
Switched to a new branch 'nueva-rama'
```

Para probar otro uso del comando “git checkout”, restauré el archivo git.txt a su última versión confirmada en el repositorio, que al no devolver nada este comando, indicó que el archivo se restauró correctamente y ahora está en su estado más reciente confirmado.

```
PS C:\Users\34636\OneDrive - Universidad Pontificia Comillas\Escritorio\PAT\Prácticas\p1> git checkout -- git.txt
```