

Git es una herramienta de control de versiones. Permite controlar el proceso de creación de software llevando un registro exhaustivo de todos los cambios realizados. Ofrece la posibilidad de crear versiones, de ramificar un proyecto en diferentes flujos e incluso de volver hacia atrás deshaciendo los últimos cambios realizados. Git es un programa libre y gratuito que se distribuye mediante la licencia GNU (GPL 2.0).

En este ejercicio vamos a tratar de familiarizarnos con sus funciones básicas:

- Instalemos Git en nuestro PC: <https://git-scm.com/downloads>
- Comprobemos la instalación con 'git --version'
- Creemos un directorio para este ejercicio y añadamos algunos ficheros de texto en él
- Creemos nuestra propia cuenta en Github
- Crea un repositorio desde Github (preferentemente con el mismo nombre que tu directorio)
- Abre una terminal y configura Git para añadir el nombre de usuario de Github: `git config --global user.name <nombre usuario Github>`
- Añade también el email de usuario de Github: `git config --global user.email <email Github>`
- Establece el método de guardado de las credenciales:
 - a. Temporal, pero más seguro: `git config --global credential.helper 'cache --timeout <seconds>'`
 - b. Permanente, pero más inseguro: `git config --global credential.helper 'store'`
- Establece el método para aportar credenciales: `git config --global credential.useHttpPath true`
- Verifica tu configuración: `git config --list`

A continuación:

- Inicialicemos desde consola nuestro repositorio: 'git init'
- Consulta el estado del "árbol de trabajo" con 'git status' y haz captura de pantalla

```
Simbolo del sistema
Microsoft Windows [Versión 10.0.22631.4169]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\nocturno>cd Desktop

C:\Users\nocturno\Desktop>cd Practica Github
El sistema no puede encontrar la ruta especificada.

C:\Users\nocturno\Desktop>cd Practica Github

C:\Users\nocturno\Desktop\Practica Github>git init
Initialized empty Git repository in C:/Users/nocturno/Desktop/Practica Github/.git/

C:\Users\nocturno\Desktop\Practica Github>git status
On branch master

No commits yet

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

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

C:\Users\nocturno\Desktop\Practica Github>
```

- Notifiquemos a Git qué ficheros queremos añadir al repositorio: 'git add . --all'
- Consulta el estado y haz captura

```
Simbolo del sistema

C:\Users\nocturno>cd Desktop

C:\Users\nocturno\Desktop>cd Practica Github
El sistema no puede encontrar la ruta especificada.

C:\Users\nocturno\Desktop>cd Practica Github

C:\Users\nocturno\Desktop\Practica Github>git init
Initialized empty Git repository in C:/Users/nocturno/Desktop/Practica Github/.git/

C:\Users\nocturno\Desktop\Practica Github>git status
On branch master

No commits yet

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

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

C:\Users\nocturno\Desktop\Practica Github>git add
Nothing specified, nothing added.
hint: Maybe you wanted to say 'git add .'
hint: Disable this message with "git config advice.addEmptyPathsSpec false"

C:\Users\nocturno\Desktop\Practica Github>git add .

C:\Users\nocturno\Desktop\Practica Github>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   1.txt
        new file:   2.txt

C:\Users\nocturno\Desktop\Practica Github>
```

- "Empaquetemos" o "preparemos para remitir" nuestra aportación al repositorio con 'git commit' (¡Ojo, este es nuestro primer commit!)
- Consulta el estado y haz captura

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22631.4169]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\nocturno>cd Desktop\Practica Github

C:\Users\nocturno\Desktop\Practica Github>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   1.txt
        new file:   2.txt

C:\Users\nocturno\Desktop\Practica Github>git commit -m "Creacion de dos archivos de texto vacios"
[master (root-commit) 04692e4] Creacion de dos archivos de texto vacios
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 1.txt
 create mode 100644 2.txt

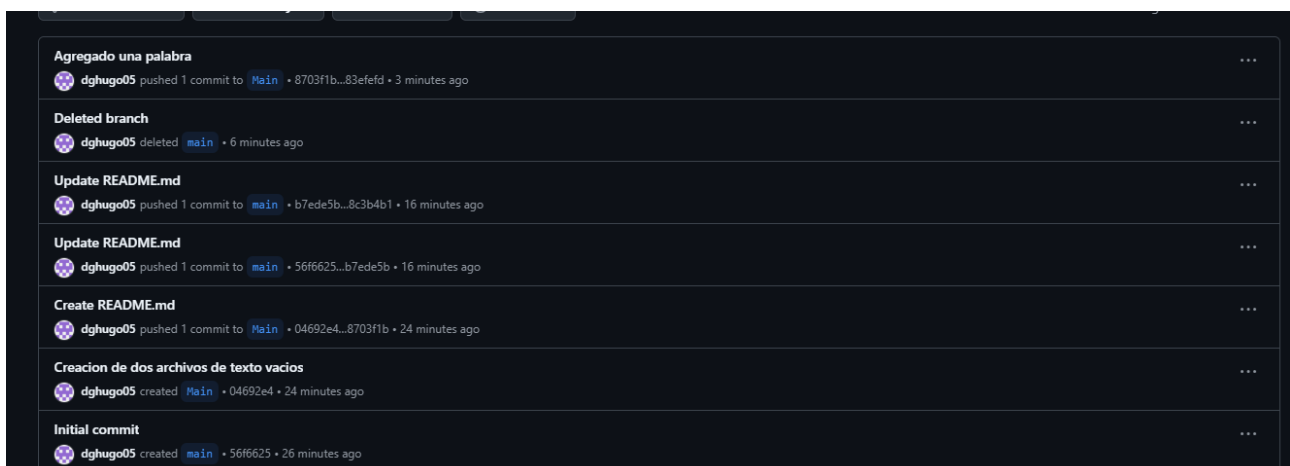
C:\Users\nocturno\Desktop\Practica Github>git status
On branch master
nothing to commit, working tree clean

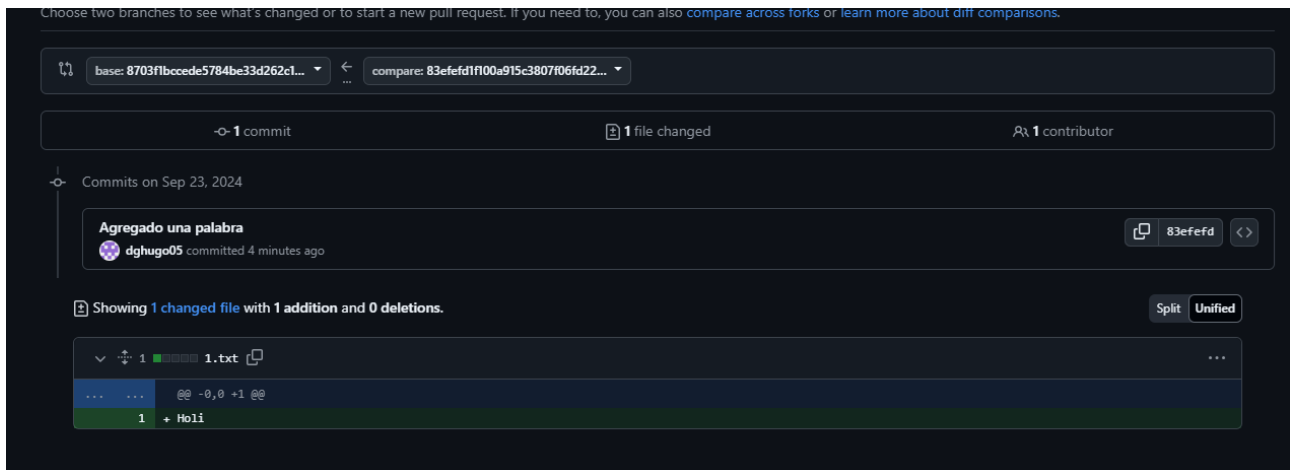
C:\Users\nocturno\Desktop\Practica Github>
```

- Cambia la rama sobre la que trabajará Git de 'master' (por defecto) a 'main' (que es la que usa github)
- Comunica a la herramienta Git la localización de tu repositorio en la nube (Github)
- Sube tus paquete a Github (git push) y haz captura. Puede que se te abra una ventana del navegador para que aporte tus credenciales.

```
C:\Users\nocturno\Desktop\Practica Github>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 6 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 302 bytes | 302.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/dghugo05/Practica-.git
 8703f1b..83efefd  Main -> Main
```

- Comprueba en Github la operación y haz una nueva captura





Cuando termines:

- Haz modificaciones manualmente a los ficheros de texto **locales**
- Consulta el estado del "árbol de trabajo" con 'git status' y haz captura de pantalla

```
Símbolo del sistema
C:\Users\nocturno\Desktop\Practica Github>git commit -m "Agregado una palabra"
[Main 83efef] Agregado una palabra
1 file changed, 1 insertion(+)

C:\Users\nocturno\Desktop\Practica Github>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 6 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 302 bytes | 302.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/dghugo05/Practica-.git
8703f1b..83efefd Main -> Main

C:\Users\nocturno\Desktop\Practica Github>git status
On branch Main
Your branch is up to date with 'origin/Main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   2.txt

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

no changes added to commit (use "git add" and/or "git commit -a")

C:\Users\nocturno\Desktop\Practica Github>
```

- Notifica a Git los ficheros a añadir al repositorio: 'git add'
- Consulta el estado y haz captura

```
C:\Users\nocturno\Desktop\Practica Github>git add 2.txt

C:\Users\nocturno\Desktop\Practica Github>
```

```

C:\Users\nocturno\Desktop\Practica Github>git add 2.txt

C:\Users\nocturno\Desktop\Practica Github>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)
        modified:   2.txt

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

C:\Users\nocturno\Desktop\Practica Github>

```

- "Empaquetemos" o "preparemos para remitir" nuestra aportación al repositorio con 'git commit'
- Consulta el estado y haz captura

```

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   2.txt

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

C:\Users\nocturno\Desktop\Practica Github>git commit -m "Añadido una frase"
[Main 7aae2f0] Añadido una frase
1 file changed, 1 insertion(+)

C:\Users\nocturno\Desktop\Practica Github>git status
On branch Main
Your branch is ahead of 'origin/Main' by 1 commit.
  (use "git push" to publish your local commits)

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

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

C:\Users\nocturno\Desktop\Practica Github>

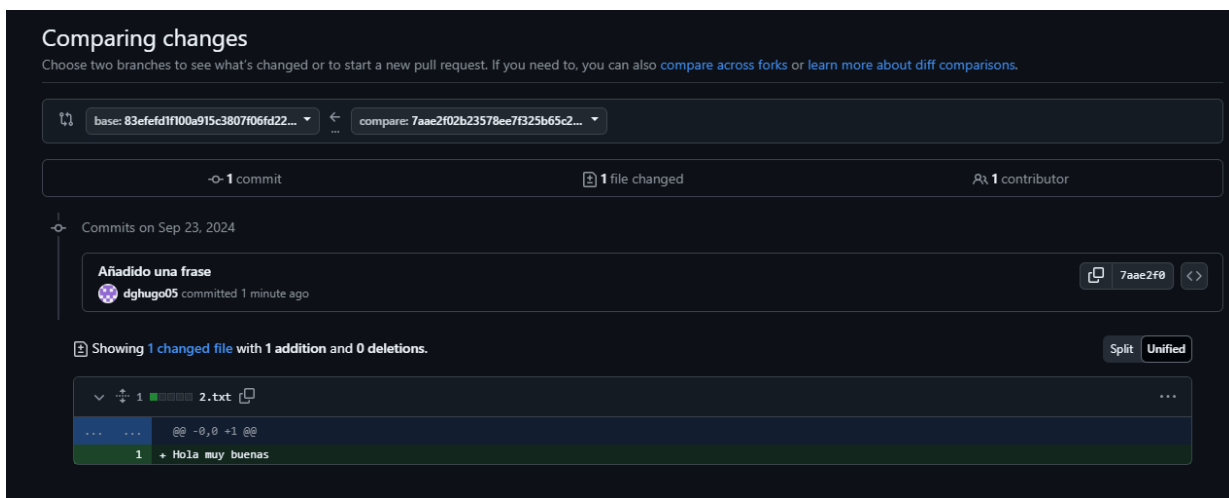
```

- Sube tus paquete a Github (git push) y haz captura

```
C:\Users\nocturno\Desktop\Practica Github>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 6 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/dghugo05/Practica-.git
83efefd..7aae2f0  Main -> Main

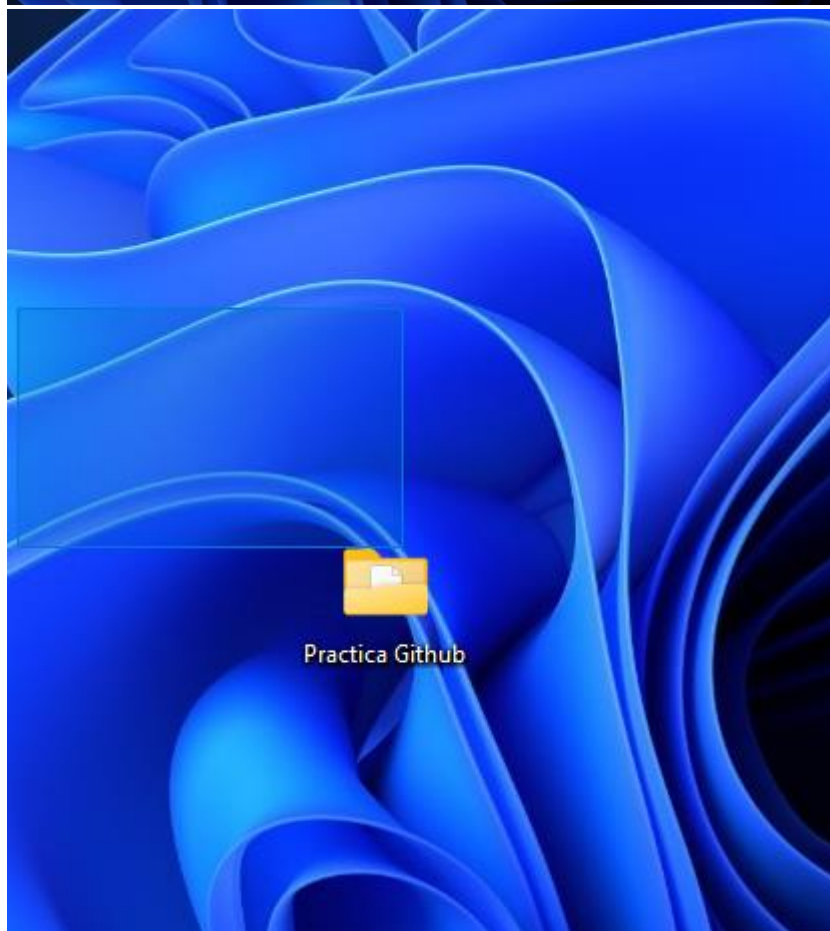
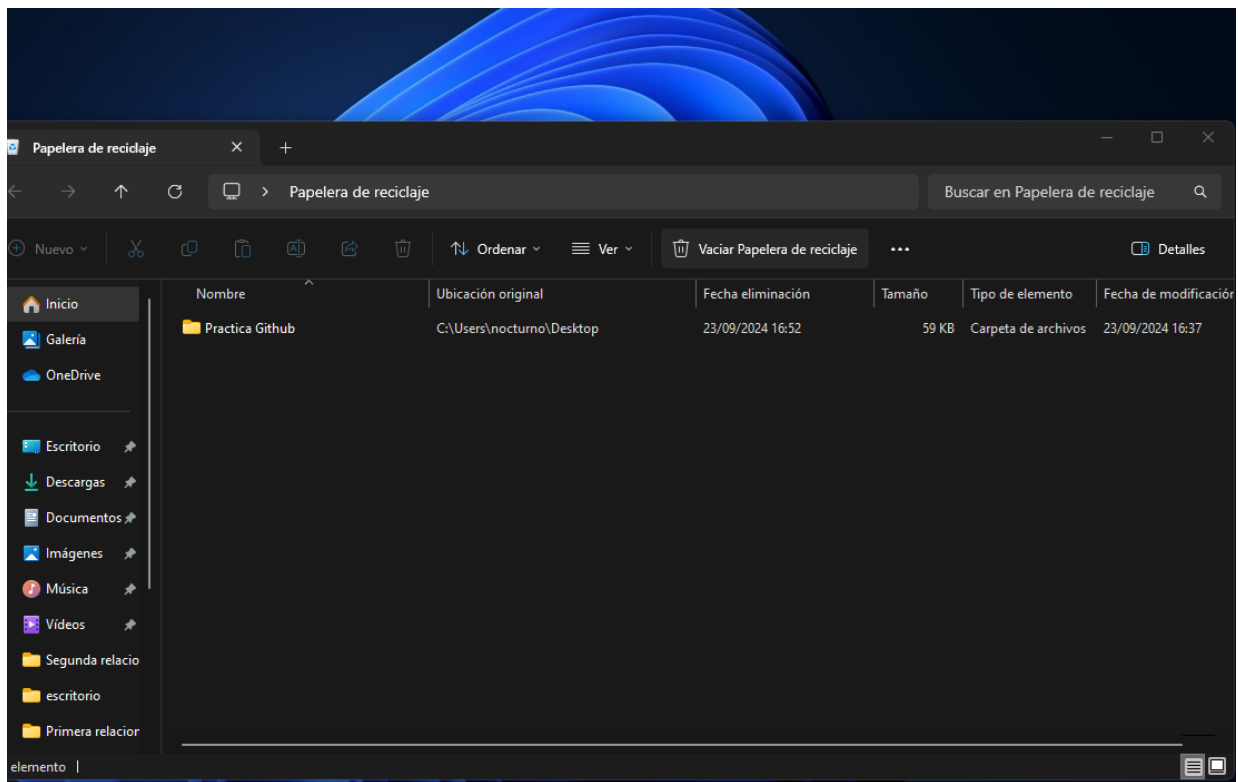
C:\Users\nocturno\Desktop\Practica Github>
```

- Comprueba en Github la operación y haz una nueva captura



Por último:

- Borra completamente el directorio **local** de trabajo (con sus ficheros)
- Recupéralo desde tu repositorio en Github
- Haz una captura de pantalla

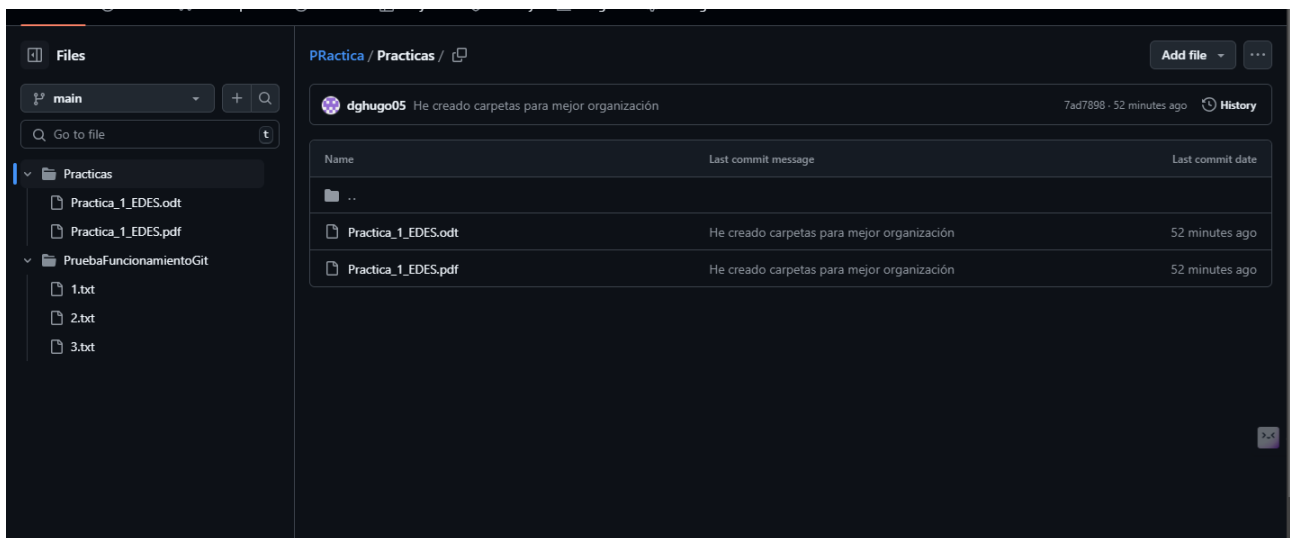


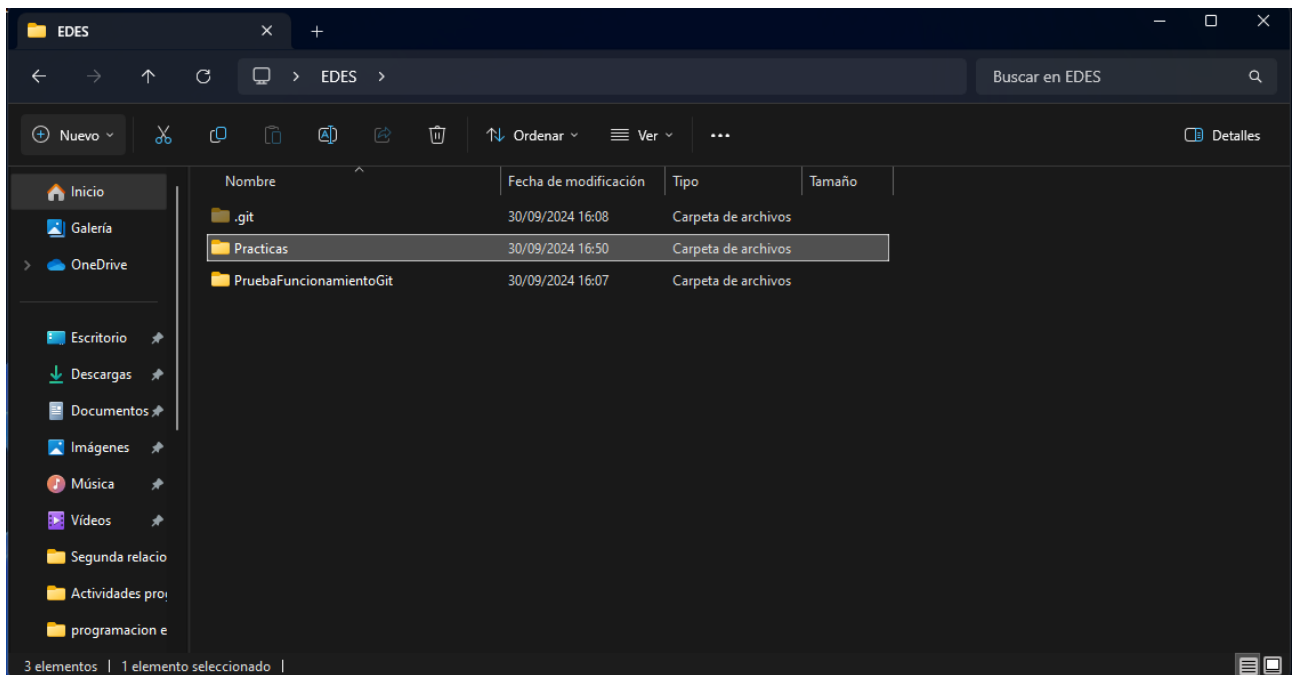
```
C:\Users\nocturno\Desktop>git pull https://github.com/dghugo05/Practica-.git Main
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 12 (delta 1), reused 8 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (12/12), 1.61 KiB | 53.00 KiB/s, done.
From https://github.com/dghugo05/Practica-
 * branch                Main                -> FETCH_HEAD

C:\Users\nocturno\Desktop>
```

Prepara una memoria bien estructurada con los comandos empleados, tus comentarios, experiencias y soluciones a los problemas encontrados. Incluye capturas de pantalla significativas.

Tengo que aclarar que estas son las capturas de un fichero que no se llama como acordamos en clase, ya que lo empecé antes, luego ya lo he repetido con los nombres acordados en clase y actualmente tengo creado tanto el repositorio local como el de Github con los nombres puestos. También cabe aclarar, que tuve ciertos problemas a la hora de repetirlo y tuve que usar unos comandos no vistos en clase como puede ser: “git push –set upstream <rama> <url>”: El cual sube los cambios y asigna el repositorio como predeterminado para subir los cambios al repositorio remoto; “git fetch”, el cual es como git pull, pero solo copia los cambios a el repositorio local, no como “git pull” que lo copia a tu directorio de trabajo. Aquí presento imágenes del repositorio realizado con los nombres propuestos en clase:





He de aclarar también que no es mi primera vez usando Github, pero si he tenido varios problemas y he tenido que rebuscar en comandos de git para encontrar una solución. El principal problema que tuve fue el de no dejarme subí archivos a mi repositorio remoto. Lo solucione principalmente con los comandos ubicados arriba, pero hubo mas comandos que use en mi travesia al intentar solucionar el problema.