

# Primer Proyecto GitHub



**git**





**GitHub**


# 1. Instalación de Git

Nos tenemos que ir a este link: <https://git-scm.com/downloads> y le damos clic a Download for windows.


## Downloads

 **macOS**

 **Windows**

 **Linux/Unix**

Older releases are available and the Git source repository is on GitHub.



### GUI Clients

Git comes with built-in GUI tools (**git-gui**, **gitk**), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients →](#)

### Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

[View Logos →](#)

### Git via Git

If you already have Git installed, you can get the latest development version via Git itself:

```
git clone https://github.com/git/git
```

You can also always browse the current contents of the git repository using the [web interface](#).

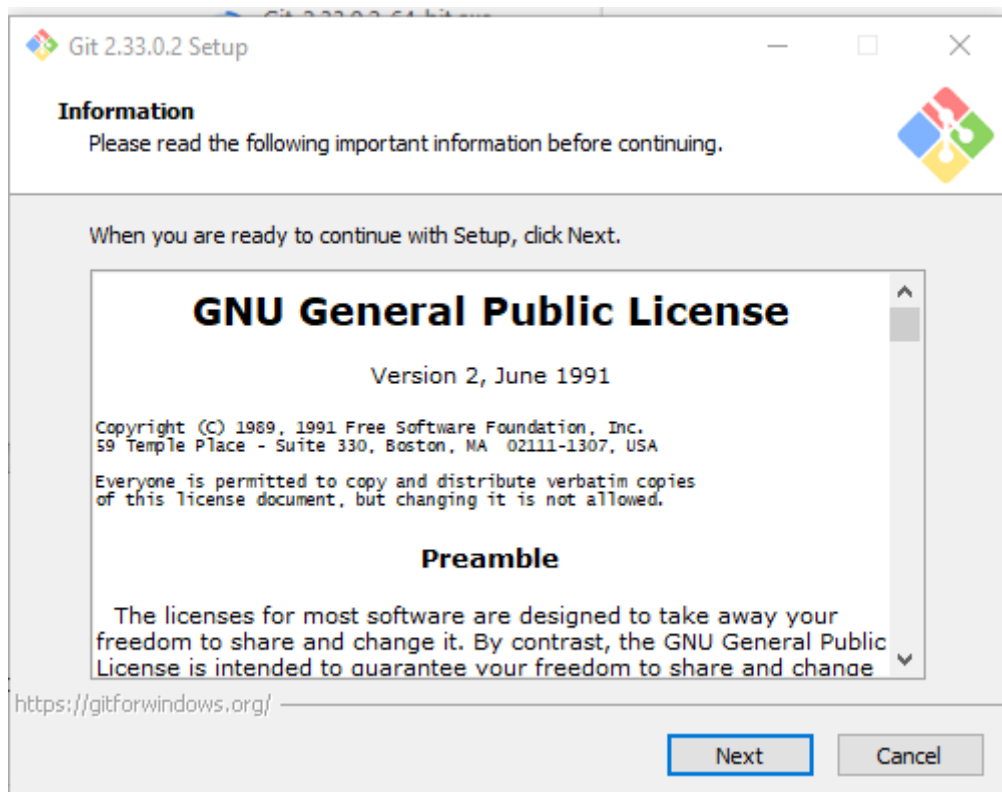


Git-2.33.0.2-64-bit.exe  
30,0/47,8 MB, Quedan 2 s

Vamos a la carpeta donde se ha descargado y le damos clic.

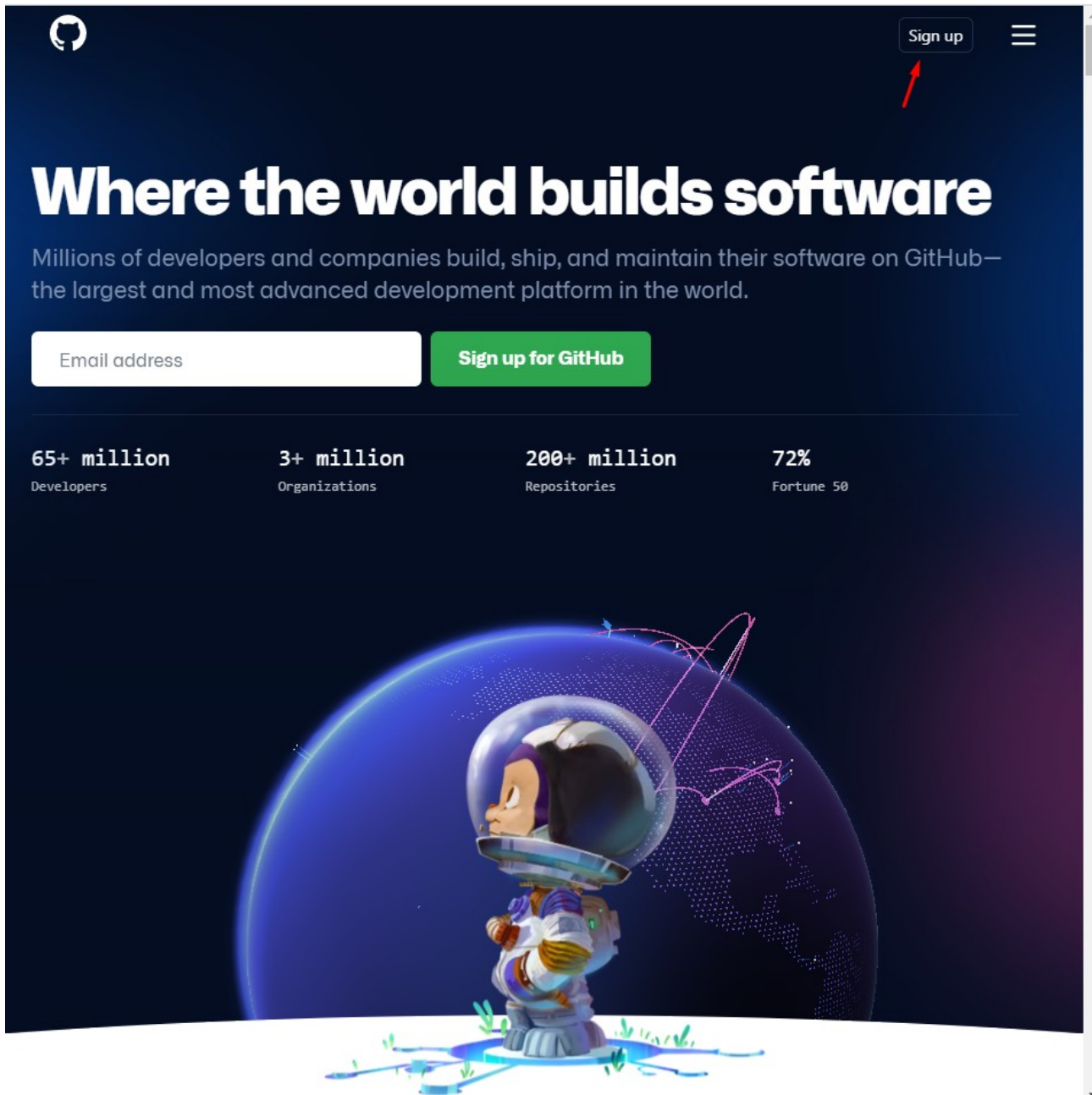


Básicamente le damos todo a siguiente y por defecto.



## 2. Crear cuenta de Github

Nos tenemos que ir a este link: <https://github.com/> y le damos clic a Sign up y básicamente nos creamos una cuenta.



Iniciamos sesión



Sign in to GitHub

Username or email address

ugartepenamanuel

Password

[Forgot password?](#)

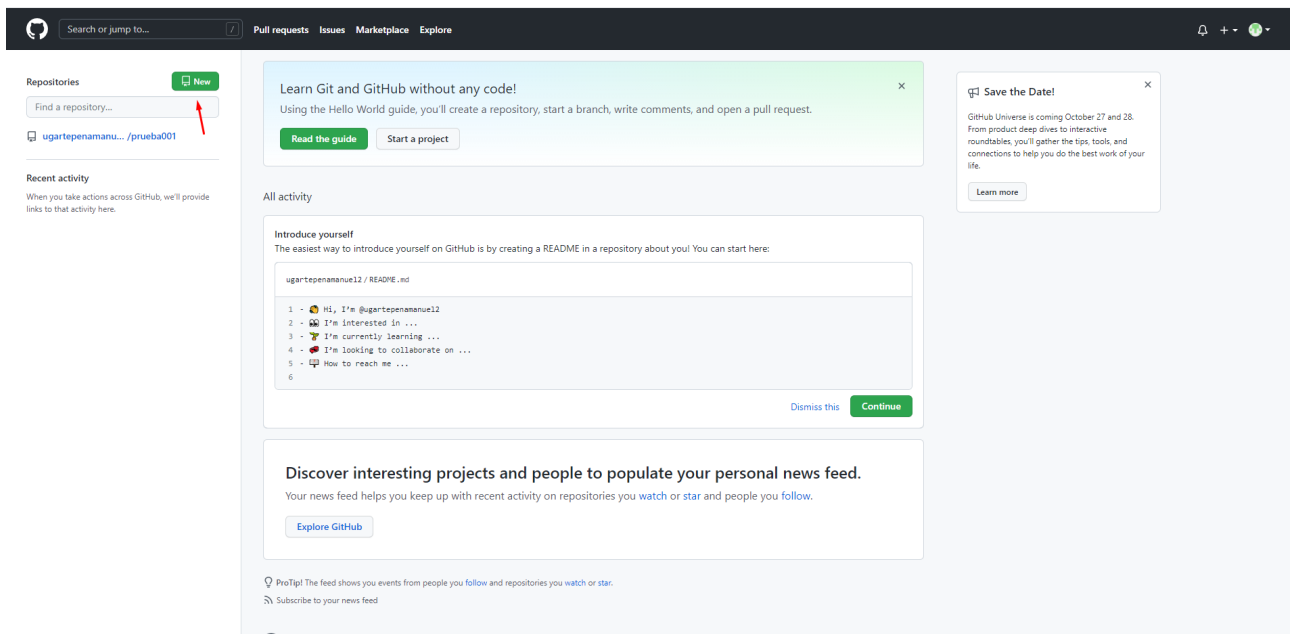
.....

Sign in

New to GitHub? [Create an account.](#)

### 3. Crear nuestro primer repositorio

Le damos clic a New




En Repository name ponemos el nombre del repositorio y lo demás lo dejamos por defecto.

## Create a new repository



A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

---

Owner *	Repository name *
 ugartepenmanuel2 ▾	/ Prueba002 ✓

Great repository names are short and memorable. Need inspiration? How about [crispy-invention?](#)

Description (optional)

- 
- ☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.
- ☐  **Private**  
You choose who can see and commit to this repository.
- 

### Initialize this repository with:

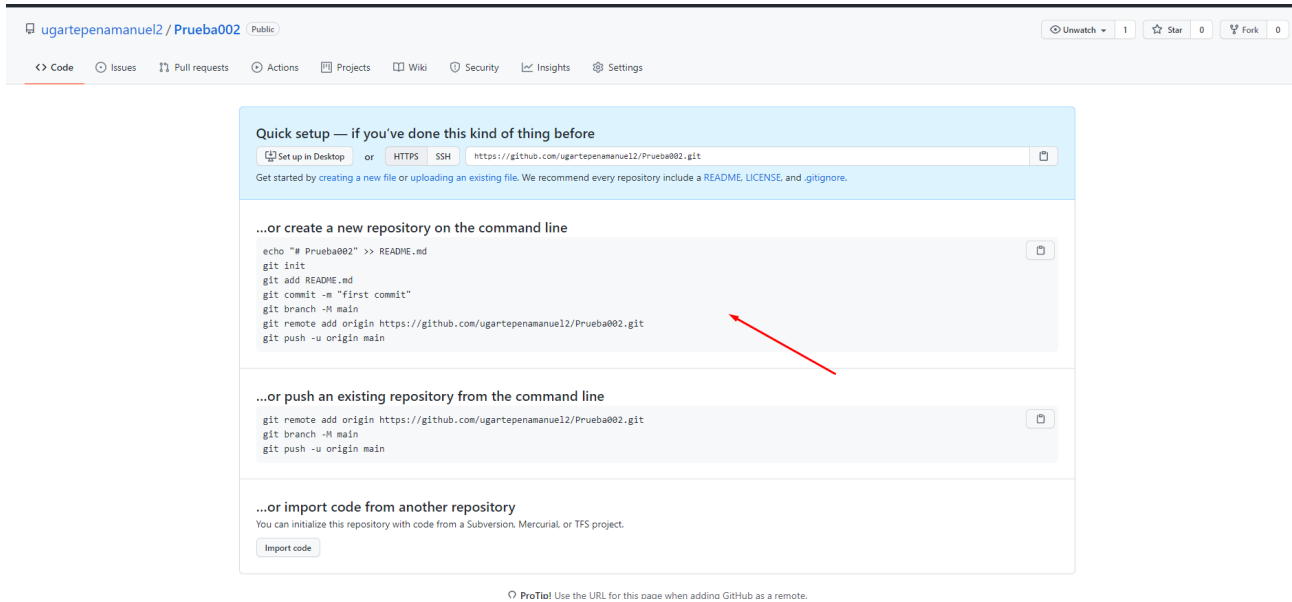
Skip this step if you're importing an existing repository.

- ☐ **Add a README file**  
This is where you can write a long description for your project. [Learn more.](#)
- ☐ **Add .gitignore**  
Choose which files not to track from a list of templates. [Learn more.](#)
- ☐ **Choose a license**  
A license tells others what they can and can't do with your code. [Learn more.](#)
- 

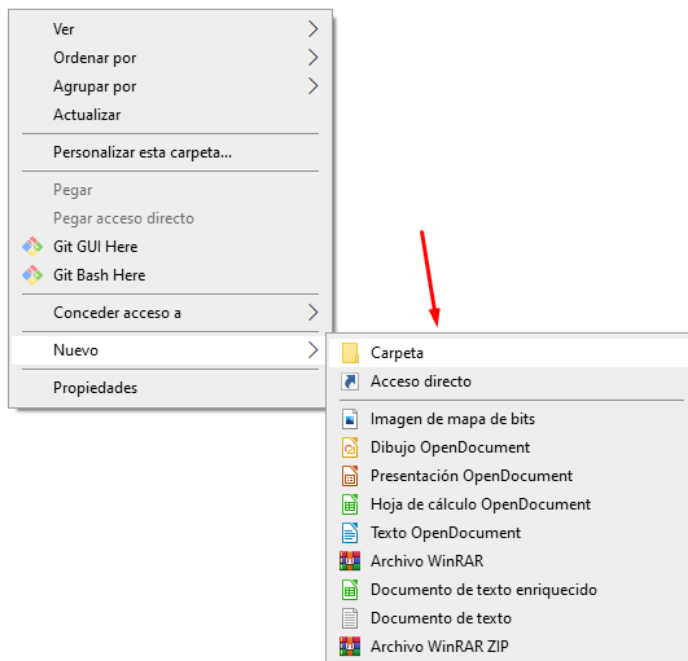
Create repository

---

En esta ventana básicamente tenemos que echarle cuenta al primero esto sirve para importar un repositorio.

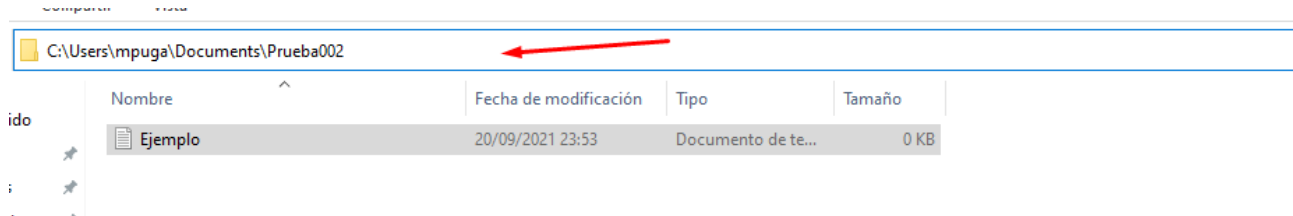


Creamos una carpeta donde queremos nuestro repositorio.





Creamos un txt por ejemplo y copiamos la ruta y nos vamos a la powershell



Iniciamos el git con el comando git init y vemos que el repositorio está vacío.

```
PS C:\Users\mpuga\Documents\Prueba002> cd C:\Users\mpuga\Documents\Prueba002
PS C:\Users\mpuga\Documents\Prueba002> git init
Initialized empty Git repository in C:/Users/mpuga/Documents/Prueba002/.git/
PS C:\Users\mpuga\Documents\Prueba002> git status
On branch master

No commits yet

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

nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\mpuga\Documents\Prueba002> █
```

Con git add Ejemplo.txt lo añadimos hacemos un git status esta todo creado.

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

PS C:\Users\mpuga\Documents\Prueba002> █
```

Lo ultimo que vamos hacer es enlazar nuestra cuenta de github con el git que se hace con los comando que tenemos en la captura de pantalla.

```
PS C:\Users\mpuga\Documents\Prueba002> git commit -m "Prueba002"
Author identity unknown
```

A continuación tenemos que poner estos comandos con nuestros datos.

```
*** Please tell me who you are.

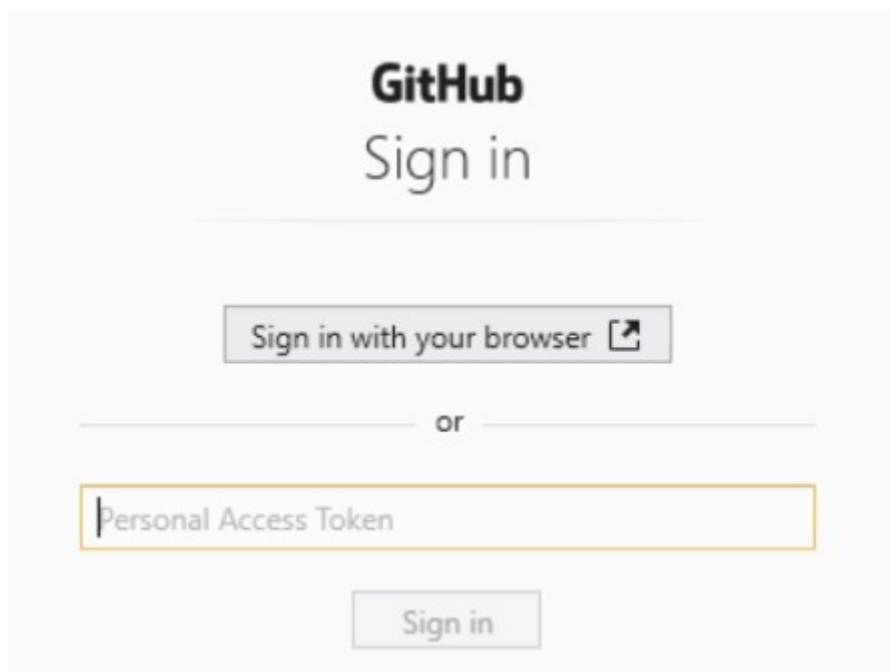
Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.




fatal: unable to auto-detect email address (got 'mpuga@DESKTOP-U9CRA3L.(none)')
PS C:\Users\mpuga\Documents\Prueba002> git config --global user.email "mugapen809g.educaand.es"
PS C:\Users\mpuga\Documents\Prueba002> git config --global user.name "ugartepenamanuel12"
PS C:\Users\mpuga\Documents\Prueba002> git commit -m "Prueba002"
[master (root-commit) 8d39aa0] Prueba002
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Ejemplo.txt
PS C:\Users\mpuga\Documents\Prueba002> git branch -M main
PS C:\Users\mpuga\Documents\Prueba002> git remote add origin https://github.com/ugartepenamanuel12/Prueba002.git
PS C:\Users\mpuga\Documents\Prueba002> git push -u origin main
```

Le damos clic a Sign in with your browser



The image shows the GitHub Sign in page. At the top, it says "GitHub Sign in". Below this, there is a button labeled "Sign in with your browser" with an external link icon. Underneath the button, the word "or" is centered. Below "or", there is a text input field labeled "Personal Access Token". At the bottom of the input field, there is a "Sign in" button.


Nos aparece una ventana que tendría que salir algo parecido.





## Authentication Succeeded

You may now close this tab and return to the application.

---

 main ▾

 1 branch

 0 tags


Go to file

Add file ▾

Code ▾

ugartepenmanuel2 Prueba002

8d39aa0 5 minutes ago 1 commit

 Ejemplo.txt

Prueba002

5 minutes ago

Help people interested in this repository understand your project by adding a README.

Add a README

## 4. Uso de comandos

### Git clone

Este comando básicamente lo que permite es clonar.

Vamos a crear una carpeta llamada Copia y ahí dentro vamos a poner el repositorio.

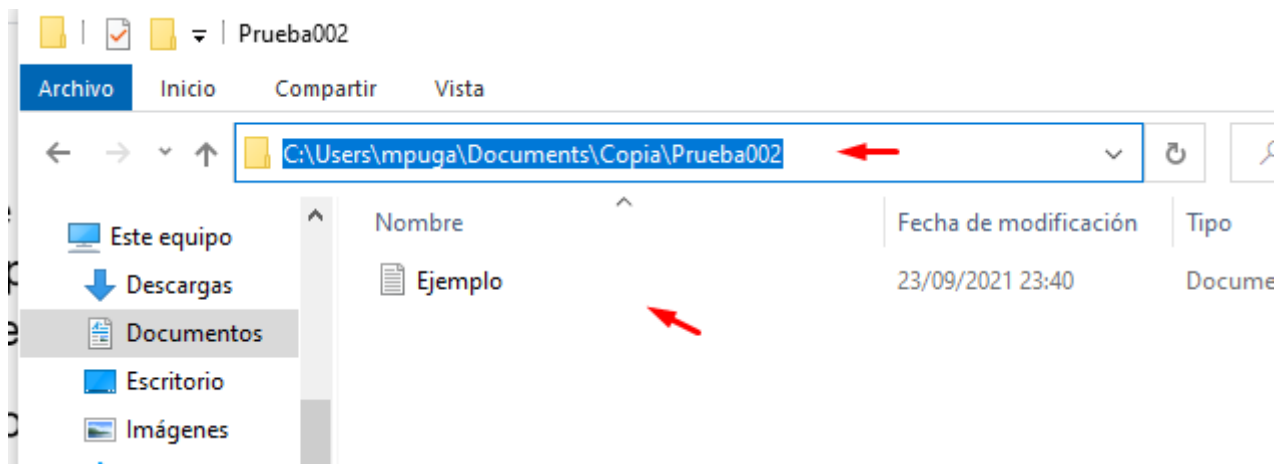
Prueba 1	23/09/2021 10:30	PERSONAL INFORMACI...	3 KB
Copia	23/09/2021 23:36	Carpeta de archivos	

Copiamos la url del repositorio que queremos clonar en este caso es Prueba002

The screenshot shows a web browser with multiple tabs. The active tab is 'ugartepenamanuel2/Prueba002'. The address bar shows the URL 'github.com/ugartepenamanuel2/Prueba002'. The page displays the GitHub repository interface for 'ugartepenamanuel2 / Prueba002'. It includes a search bar, navigation links (Pull requests, Issues, Marketplace, Explore), and repository details. The repository is public and has 1 branch (main) and 0 tags. A file 'Ejemplo.txt' is listed under the 'main' branch, committed 3 days ago. A green button 'Add a README' is visible at the bottom.

Nos metemos en la shell y entramos en la carpeta donde queremos meter el repositorio y luego ponemos el comando git clone url.

```
PS C:\Users\mpuga\Documents\Prueba002> cd C:\Users\mpuga\Documents\Copia
PS C:\Users\mpuga\Documents\Copia> git init
Initialized empty Git repository in C:/Users/mpuga/Documents/Copia/.git/
PS C:\Users\mpuga\Documents\Copia> git clone https://github.com/ugartepenamanuel12/Prueba002
Cloning into 'Prueba002'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
PS C:\Users\mpuga\Documents\Copia>
```

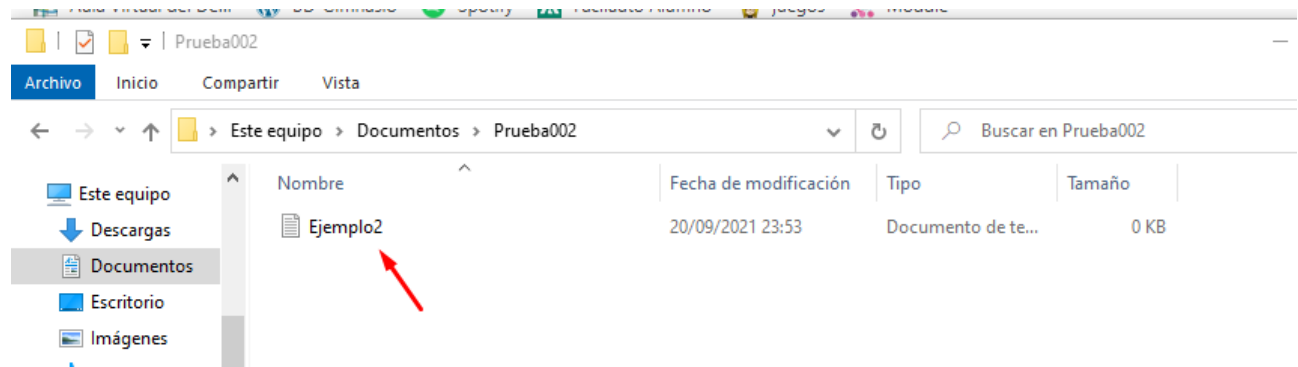


## Git mv

Este comando permite mover o cambiar de nombre del archivo en nuestro caso vamos a moverlo a mover.

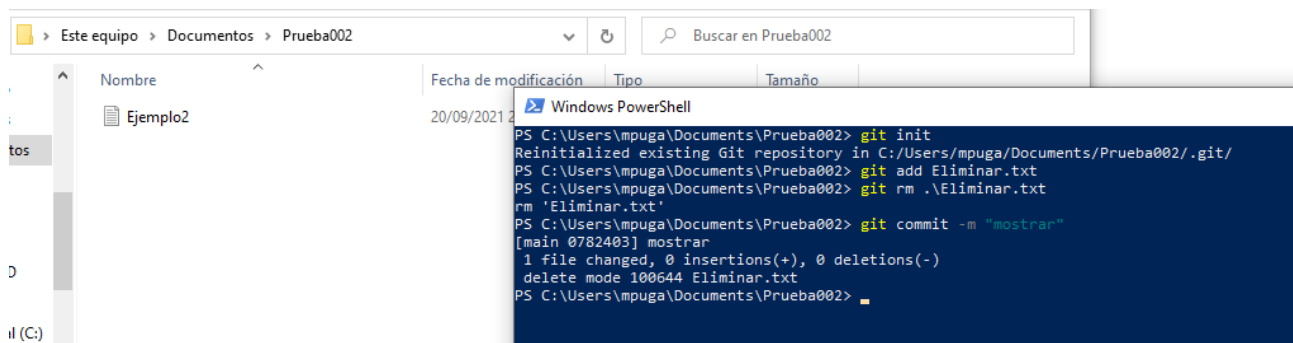
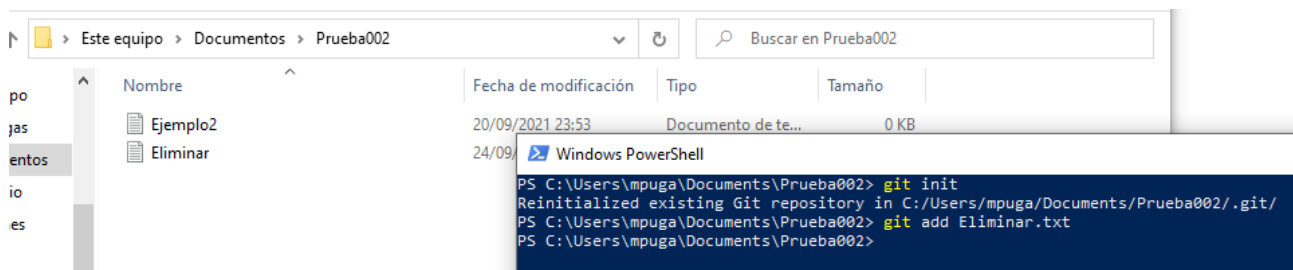
Para cambiar el nombre tenemos que metemos en la shell y utilizamos git mv .\ como\_se\_llama\_el\_archivo nuevo\_nombre

```
PS C:\Users\mpuga\Documents\Copia> cd C:\Users\mpuga\Documents\Prueba002
PS C:\Users\mpuga\Documents\Prueba002> git mv .\Ejemplo.txt Ejemplo2.txt
PS C:\Users\mpuga\Documents\Prueba002>
```



## Git rm

Este comando permite eliminar repositorios.



## Git notes

Este comando permite poner notas.

```
Prueba002
PS C:\Users\mpuga\Documents\Prueba002> git notes add b1120d2149d3d4e2133617710a717cc2d69698de -m "ejemplo"
PS C:\Users\mpuga\Documents\Prueba002> git log b1120d2149d3d4e2133617710a717cc2d69698de
commit b1120d2149d3d4e2133617710a717cc2d69698de
Author: ugartepenamanuel12 <mugapen809g.educaand.es>
Date:   Fri Sep 24 00:10:20 2021 +0200

    betis

Notes:
    ejemplo

commit 8d39aa0dbc4cc9e3745700ab6ceba338dbfcabec (origin/main)
Author: ugartepenamanuel12 <mugapen809g.educaand.es>
Date:   Tue Sep 21 00:12:31 2021 +0200

    Prueba002
PS C:\Users\mpuga\Documents\Prueba002>
```

## Git log

Este comando permite ver los registros.

```
PS C:\Users\mpuga\Documents\Prueba002> git log
commit 0782403dfe5d832087d773f79eb7f6976a1d931a (HEAD -> main)
Author: ugartepenamanuel12 <mugapen809g.educaand.es>
Date:   Fri Sep 24 00:13:12 2021 +0200

    mostrar

commit b1120d2149d3d4e2133617710a717cc2d69698de
Author: ugartepenamanuel12 <mugapen809g.educaand.es>
Date:   Fri Sep 24 00:10:20 2021 +0200

    betis

Notes:
    ejemplo

commit 8d39aa0dbc4cc9e3745700ab6ceba338dbfcabec (origin/main)
Author: ugartepenamanuel12 <mugapen809g.educaand.es>
Date:   Tue Sep 21 00:12:31 2021 +0200

    Prueba002
PS C:\Users\mpuga\Documents\Prueba002>
```

## .gitignore

Sirve básicamente para agregar nombre de los archivos que queremos que el repositorio ignore.

