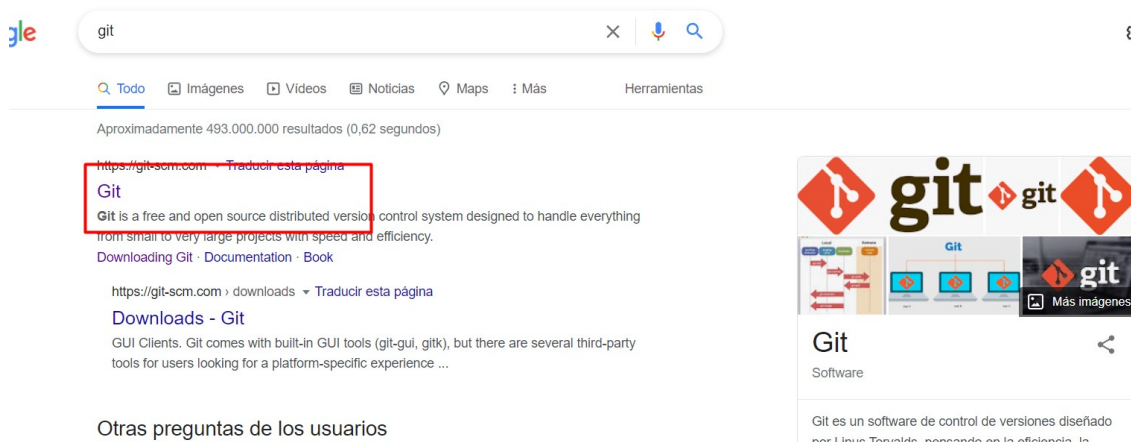
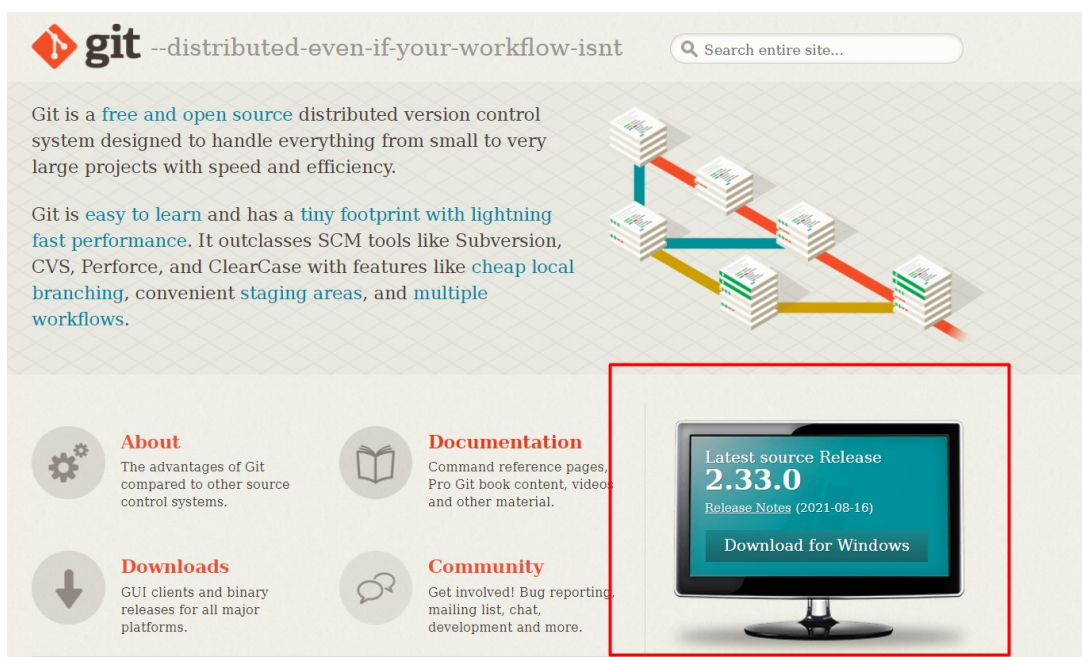


1-. Descargar Git:

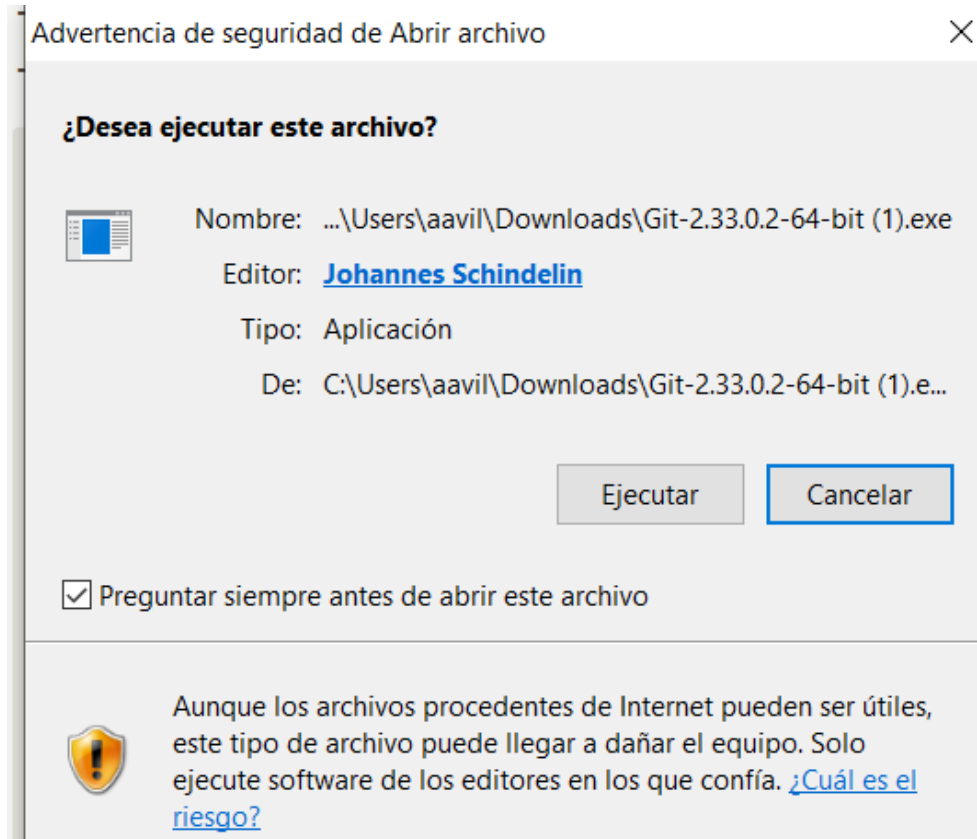
- En primer lugar descargaremos el git, buscándolo directamente en el navegador:



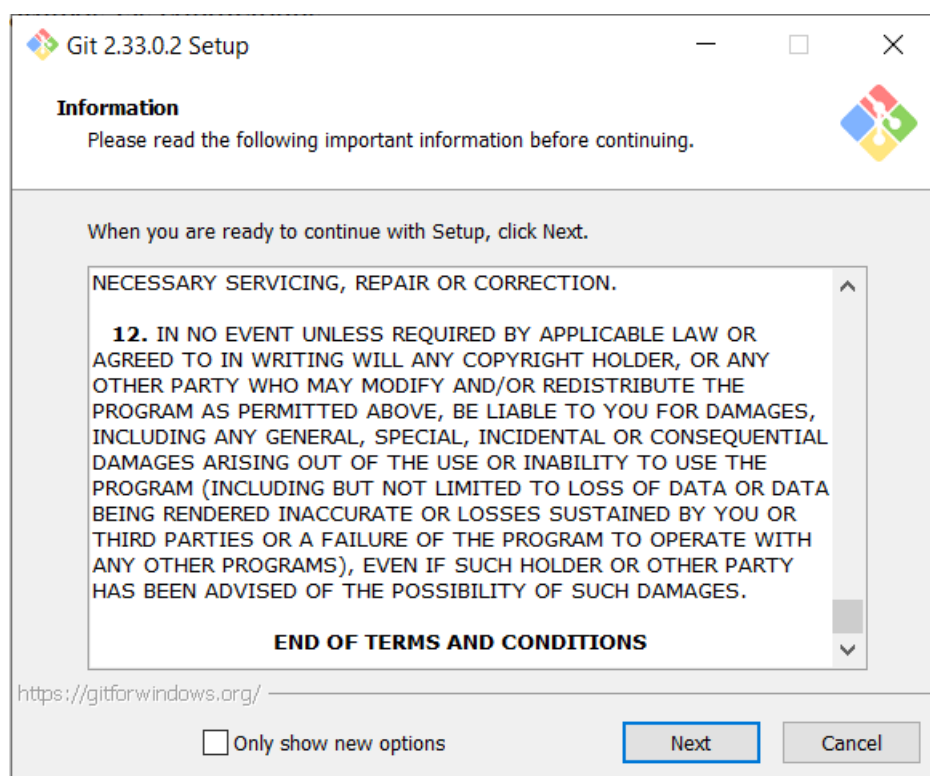
Una vez estemos dentro de la página de Git, descargaremos la última versión disponible. La descargaremos haciendo click en el monitor que nos aparece:



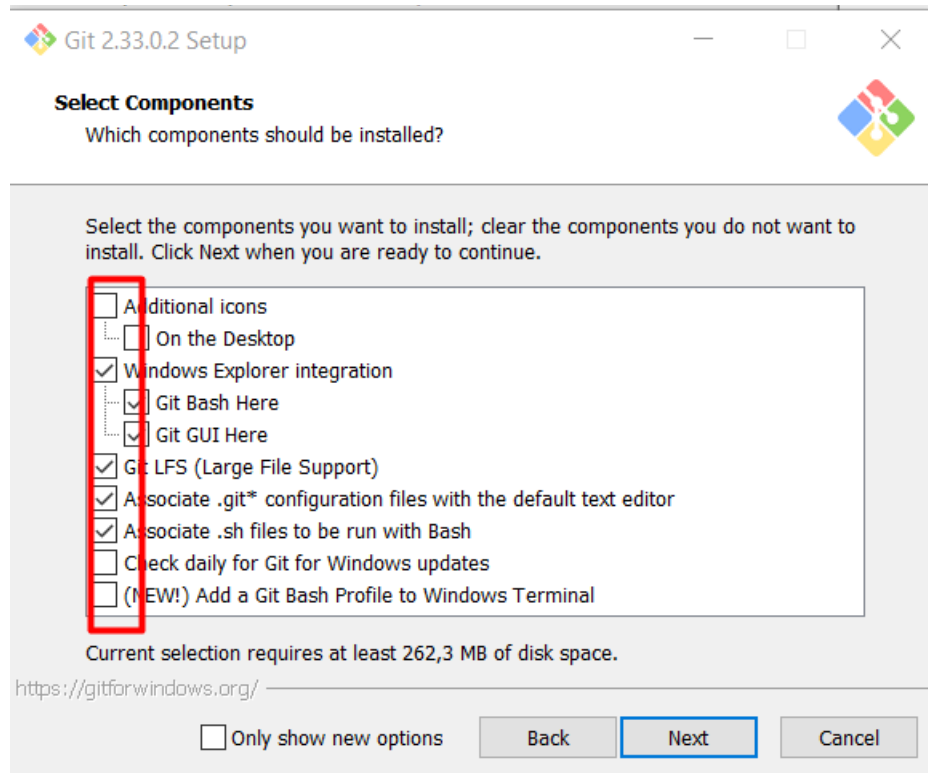
Ejecutamos el archivo de instalación:



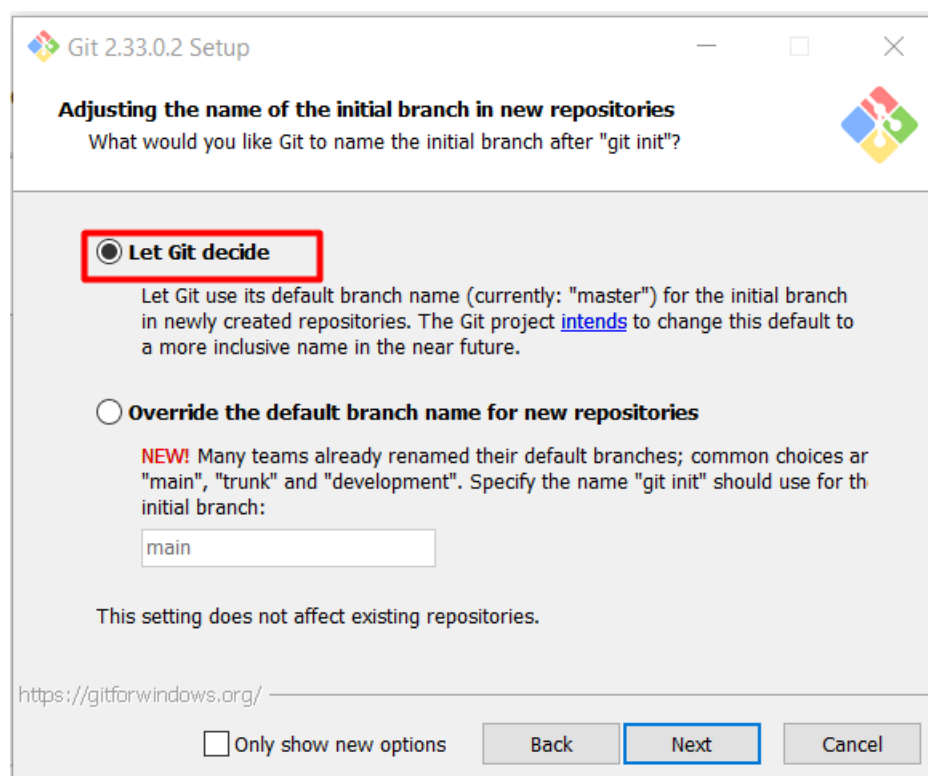
Luego aceptaremos las condiciones:



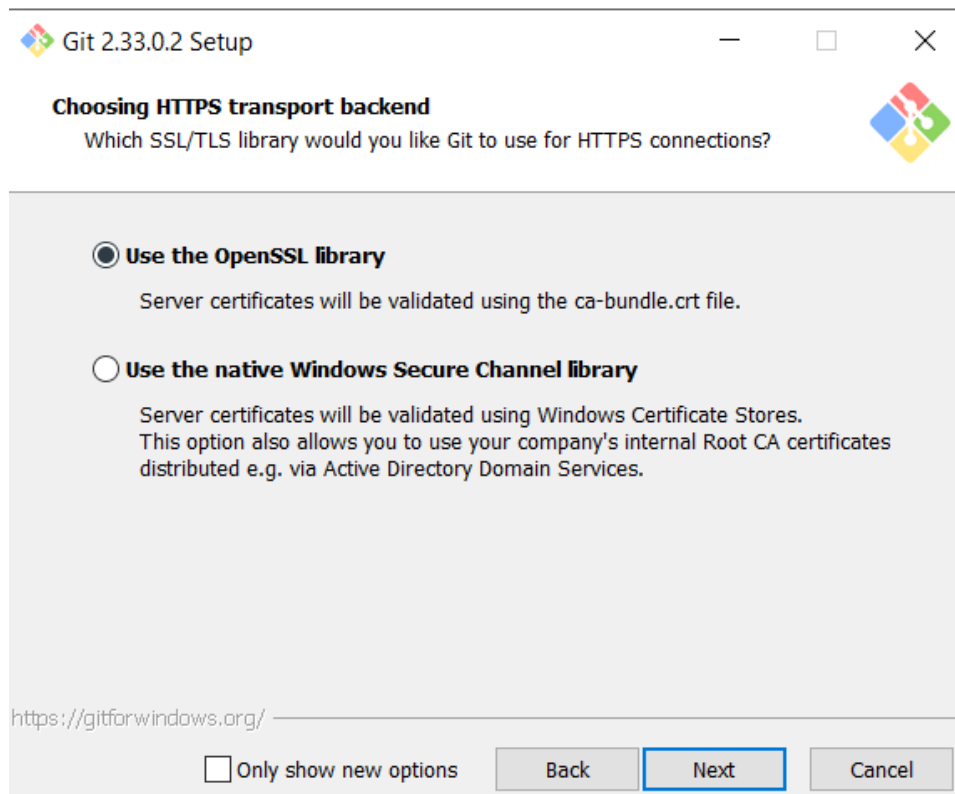
De aquí ni desmarcaremos ni marcaremos nada nuevo:



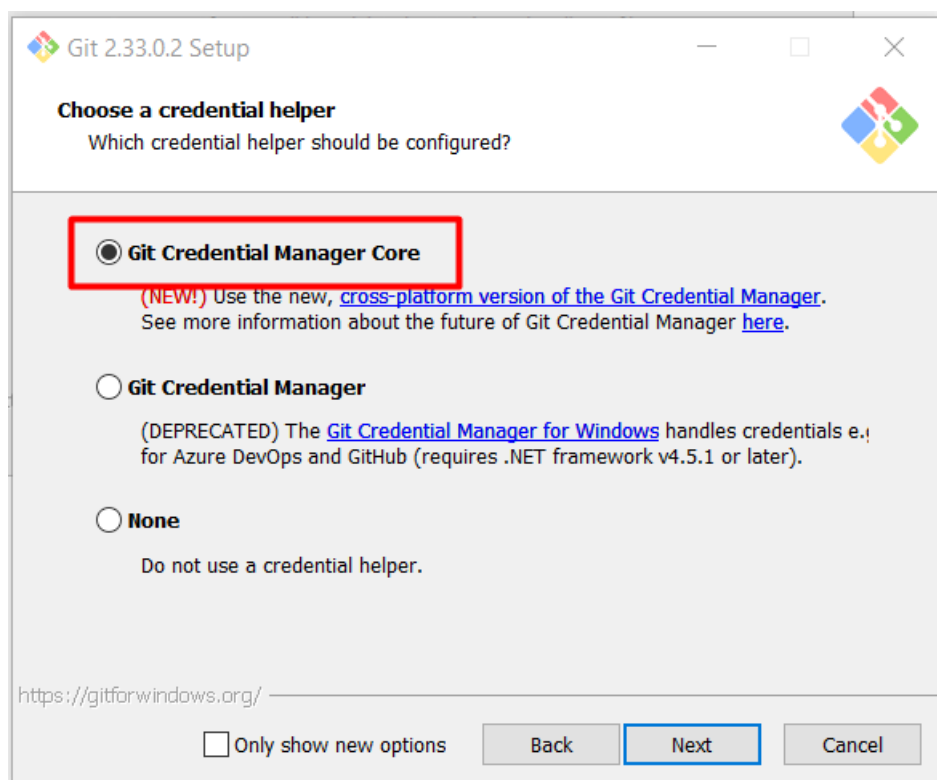
En la siguiente pantalla, marcaremos esta opción:



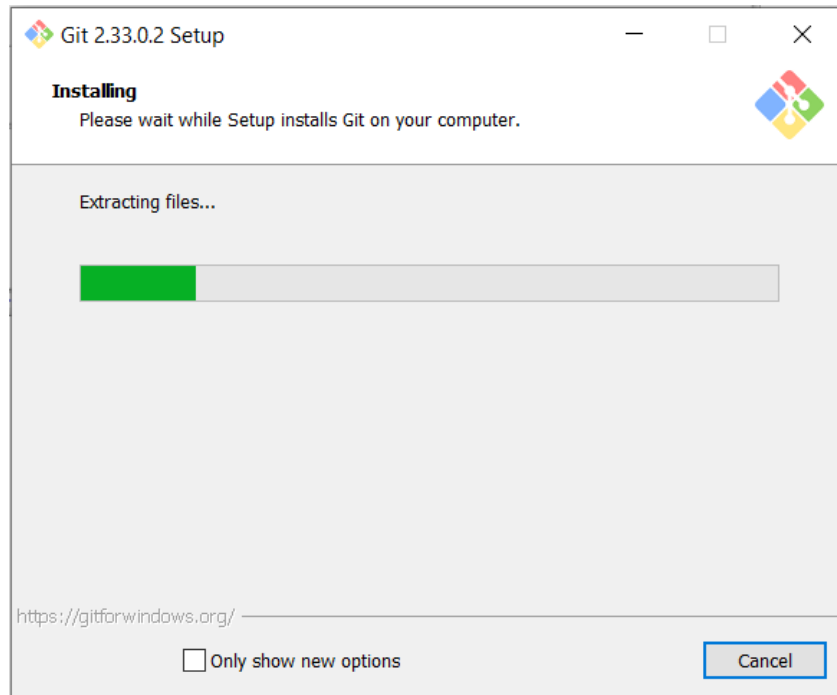
Ahora elegimos la primera opción:



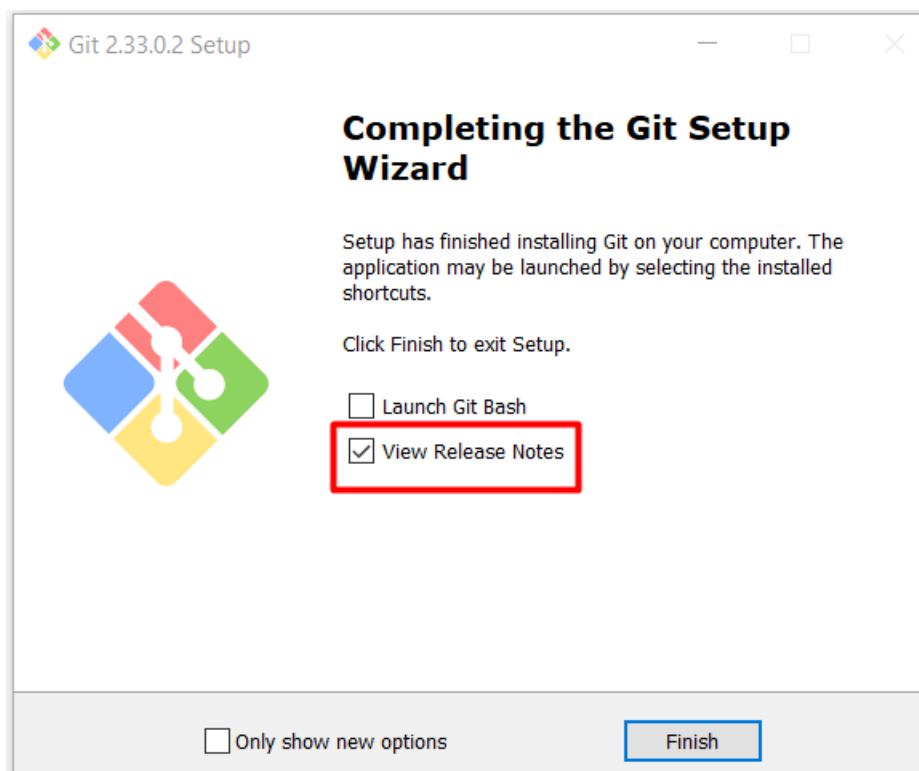
En las opciones para las credenciales marcaremos esta primera:



Ya tendríamos instalándose el git:



Y en esta opción dejaremos la predeterminada:



Cuando acabe la instalación, nos saltará al navegador con esta pantalla:

PÁGINA
PRINCIPAL
PREGUNTAS MÁS
FRECUENTES
CONTRIBUIR
INSECTOS
PREGUNTAS



Notas de la versión de Git para Windows v2.33.0 (2)

Última actualización: 24 de agosto de 2021

Introducción

Estas notas de la versión describen problemas específicos de la versión de Git para Windows. Las notas de la versión que cubren el historial de los comandos principales de git se pueden encontrar [en el proyecto Git](#) .

Consulte <http://git-scm.com/> para obtener más detalles sobre Git, incluidos los puertos a otros sistemas operativos. Git para Windows está alojado en <https://gitforwindows.org/> .

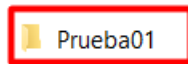
▷ Problemas conocidos

Si encuentra otros problemas, primero busque [el rastreador de errores](#) (también mire los problemas cerrados) y [la lista de correo](#) , lo más probable es que el problema ya haya sido informado. También asegúrese de usar una versión actualizada de Git para Windows (o una [compilación de instantáneas](#)

2-. Prueba con GitHub

Comenzamos con la primera prueba del GIT.

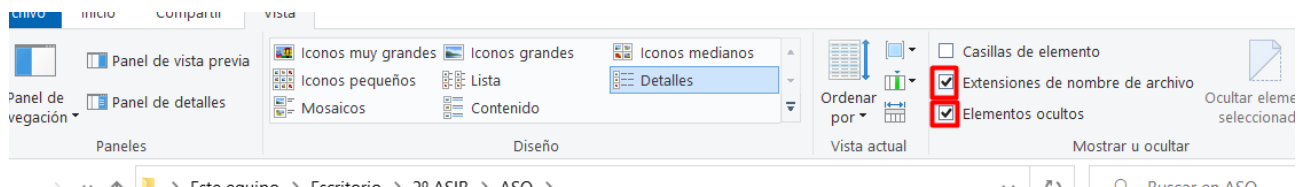
- Vamos a empezar creando una carpeta, con el nombre de prueba01 .



23/09/2021 8:37

Carpeta de archivos

Antes de seguir, haremos un par de ajustes para que nos facilite el trabajo posterior, marcaremos las dos casillas que enseñé en la siguiente captura:

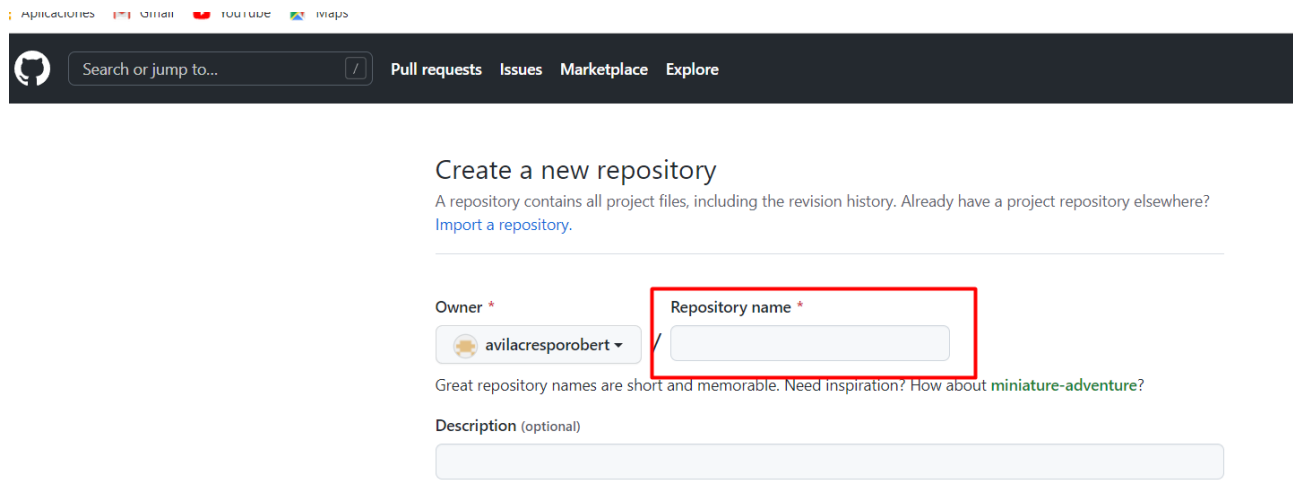


Hemos marcado la opción de que nos muestre los elementos ocultos y también que las extensiones de nombre del archivo se muestren, por ello cuando guardemos por ejemplo un archivo “txt” que se llame *HolaMundo* marcando esta opción se mostrará el archivo tal que así: *HolaMundo.txt*.

- Para comenzar la práctica creamos un archivo de texto dentro de nuestra carpeta y a continuación comenzaremos con los comandos para subirlo a la nube

3-. Crear un repositorio en GitHub

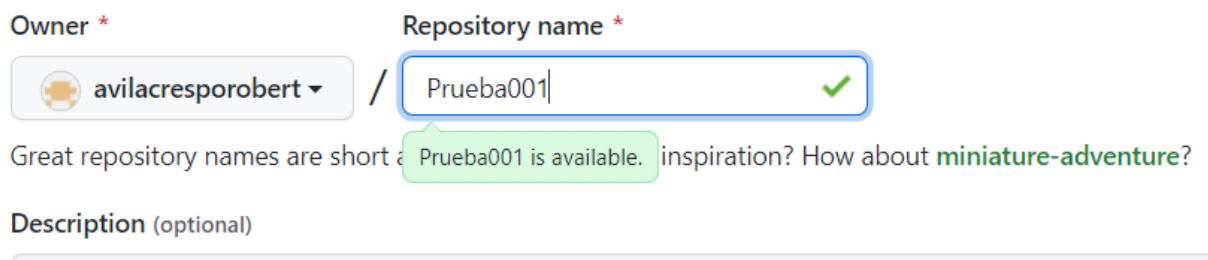
En primer lugar tendremos que crearnos un repositorio en el github, para ello previamente deberíamos de habernos creado una cuenta en github con un usuario y contraseña. Estas credenciales serán las que usemos luego en la powershell para conectar o subir nuestro archivo a la nube, concretamente a dicho repositorio.



The screenshot shows the GitHub interface for creating a new repository. At the top, there's a navigation bar with the GitHub logo, a search bar, and links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below this, the heading 'Create a new repository' is followed by a subtext: 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)'. The form has two main sections: 'Owner *' with a dropdown menu showing 'avilacresporobert' and 'Repository name *' with an empty text input field. The 'Repository name' field is highlighted with a red rectangular box. Below these fields, there's a hint: 'Great repository names are short and memorable. Need inspiration? How about [miniature-adventure?](#)'. At the bottom, there's a 'Description (optional)' label and an empty text area.

Si el nombre que le ponemos al repositorio no está anteriormente usado, nos lo proporcionará y nos dirá que podemos usarlo correctamente.

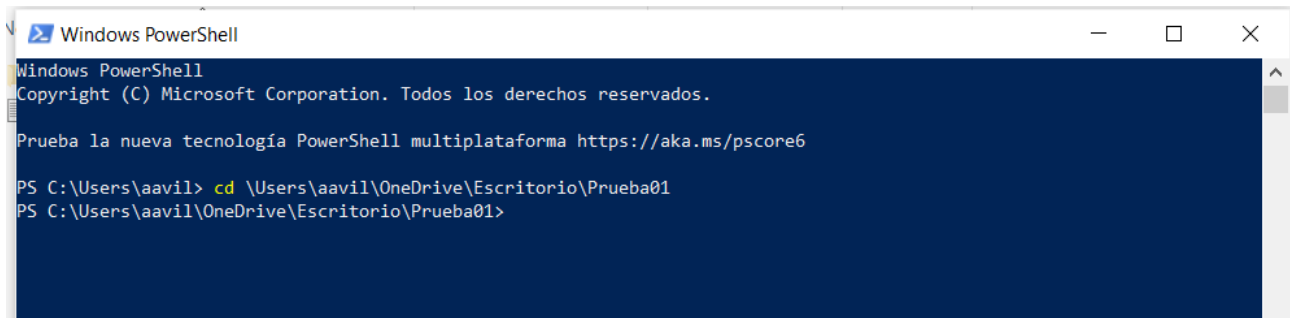
A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)



This screenshot shows the same GitHub form as above, but with the 'Repository name' field filled with the text 'Prueba001'. A green checkmark is visible at the end of the text in the input field. Below the input field, a green tooltip bubble appears with the text 'Prueba001 is available.' The rest of the form, including the 'Owner' dropdown and the 'Description' field, remains the same.

EMPEZAMOS A COMPARTIR

Lo primero que haremos es situarnos con un `cd` en dónde tenemos la carpeta que hemos creado:

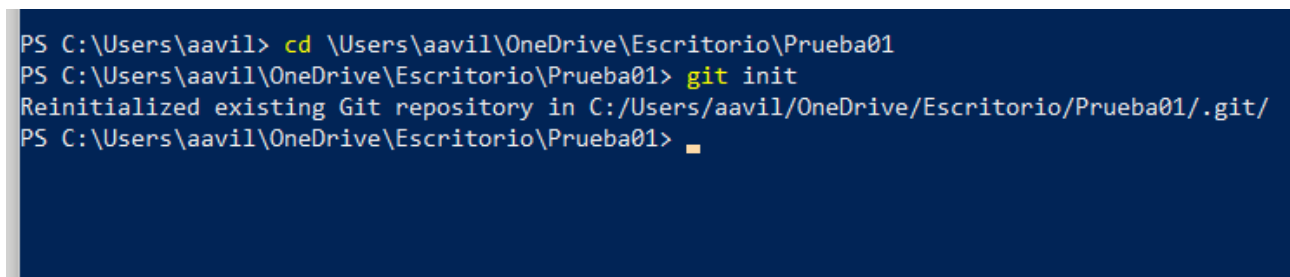


```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

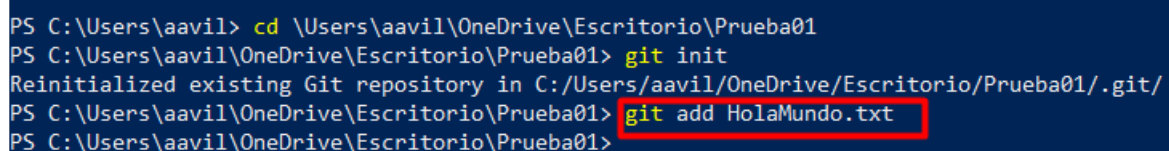
PS C:\Users\aaivil> cd \Users\aaivil\OneDrive\Escritorio\Prueba01
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01>
```

Y ahora vamos a conectar el repositorio con nuestra carpeta:



```
PS C:\Users\aaivil> cd \Users\aaivil\OneDrive\Escritorio\Prueba01
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git init
Reinitialized existing Git repository in C:/Users/aaivil/OneDrive/Escritorio/Prueba01/.git/
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01>
```

Agregaremos un archivo que en nuestro caso es HolaMundo.txt



```
PS C:\Users\aaivil> cd \Users\aaivil\OneDrive\Escritorio\Prueba01
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git init
Reinitialized existing Git repository in C:/Users/aaivil/OneDrive/Escritorio/Prueba01/.git/
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git add HolaMundo.txt
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01>
```

Actualizamos el repositorio con un comentario, utilizando el comando `git commit -m` para poder escribir el mensaje, si ponemos el comando sin el “-m” lo que hará es abrirnos un editor de texto para que escribamos nosotros ahí.

```
PS C:\Users\aaivil> cd \Users\aaivil\OneDrive\Escritorio\Prueba01
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git init
Reinitialized existing Git repository in C:/Users/aaivil/OneDrive/Escritorio/Prueba01/.git/
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git add HolaMundo.txt
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git commit -m
error: switch `m' requires a value
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git commit -m "Esta es la primera prueba"
[master (root-commit) c8b931d] Esta es la primera prueba
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 HolaMundo.txt
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> █
```

Ahora nos conectamos a nuestro repositorio:

```
PS C:\Users\aaivil> cd \Users\aaivil\OneDrive\Escritorio\Prueba01
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git init
Reinitialized existing Git repository in C:/Users/aaivil/OneDrive/Escritorio/Prueba01/.git/
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git add HolaMundo.txt
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git commit -m
error: switch `m' requires a value
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git commit -m "Esta es la primera prueba"
[master (root-commit) c8b931d] Esta es la primera prueba
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 HolaMundo.txt
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git remote add origin https://github.com/aaivilacresporobert/Prueba001.git
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> █
```

Ahora subiremos el repositorio GIT al repositorio GITHUB:



Una vez que nos hemos autenticado nos saldrá en nuestro navegador lo siguiente:

e=7031702e04ee200300400state=1400a33070704321033a3ee000301ed

Maps



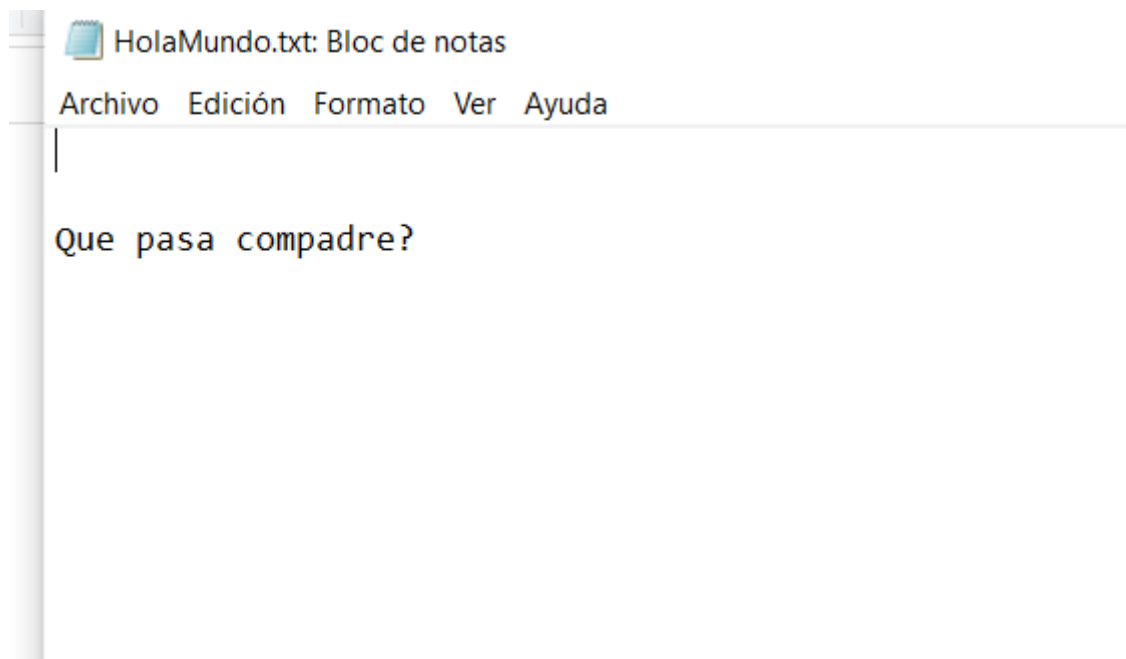
Authentication Succeeded

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

COMANDO PUSH:

Ahora probaremos a hacer un *PUSH* , este comando lo que hace es que nos sube a la repositorio git todo lo que hagamos o actualicemos en nuestro repositorio local, para comprobar que este comando, editamos el archivo txt y luego al hacer push comprobamos que también se ha cambiado en el github.

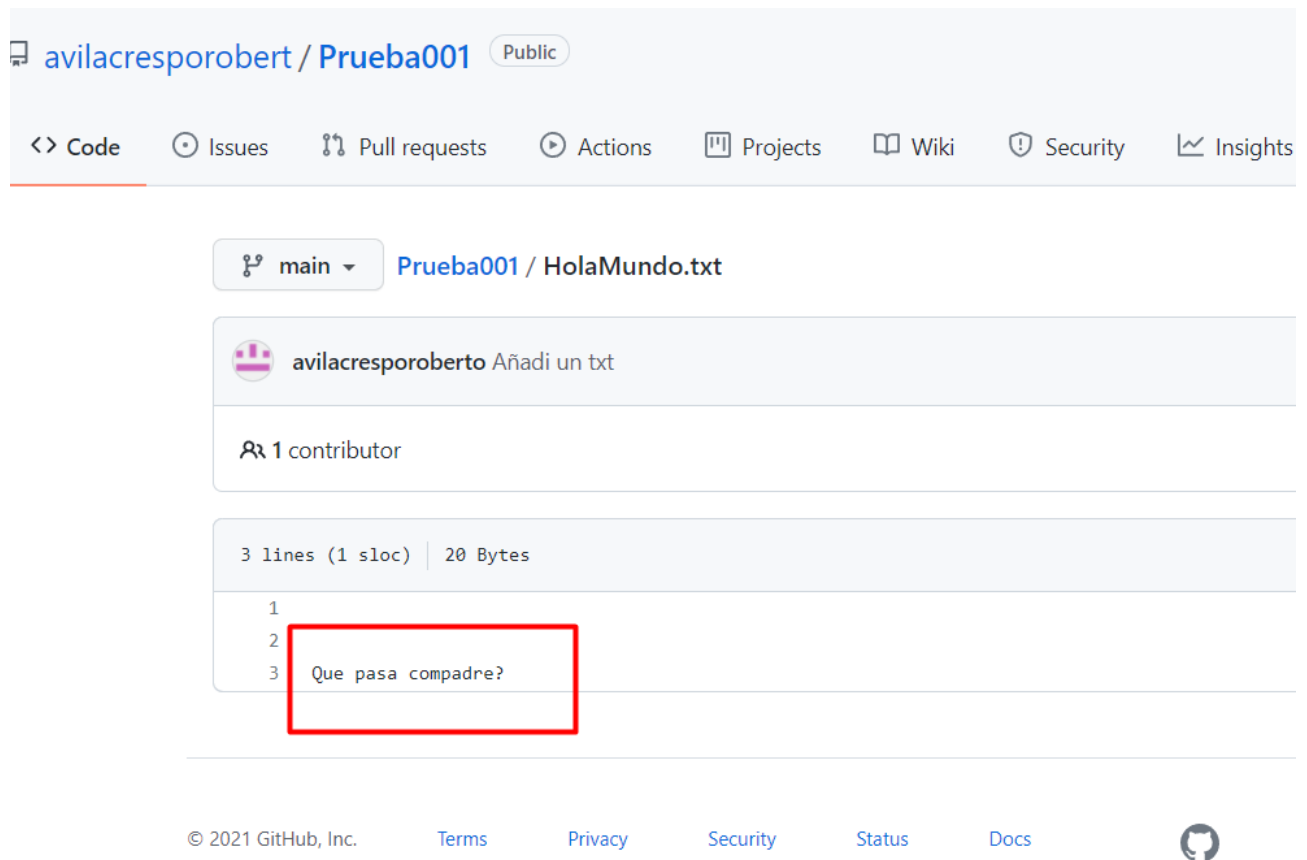
Esto sería lo que hemos escrito en nuestro txt:



Ejecutamos el push:

```
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git commit -m "Añadi un txt"
[main 2e499db] Añadi un txt
 2 files changed, 3 insertions(+)
 create mode 100644 pepito.txt
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01> git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 313 bytes | 156.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/avilacresporobert/Prueba001.git
   c8b931d..2e499db  main -> main
PS C:\Users\aaivil\OneDrive\Escritorio\Prueba01>
```

Y por último comprobamos que funciona mirando si se a subido al repositorio git el mensaje escrito dentro del txt, comprobamos que el recuadro en rojo marca lo que habíamos escrito en nuestro archivo de texto:



The screenshot shows a GitHub repository page for 'avilacresporoberto / Prueba001'. The repository is public. The main branch is selected, and the file 'Prueba001 / HolaMundo.txt' is viewed. The commit was made by 'avilacresporoberto' with the message 'Añadi un txt'. There is 1 contributor. The file details show '3 lines (1 sloc) | 20 Bytes'. The content of the file is displayed with line numbers 1, 2, and 3. Line 3 contains the text 'Que pasa compadre?', which is highlighted by a red rectangular box.

avilacresporoberto / Prueba001 Public

<> Code Issues Pull requests Actions Projects Wiki Security Insights

main ▾ Prueba001 / HolaMundo.txt

avilacresporoberto Añadi un txt

1 contributor

3 lines (1 sloc) | 20 Bytes

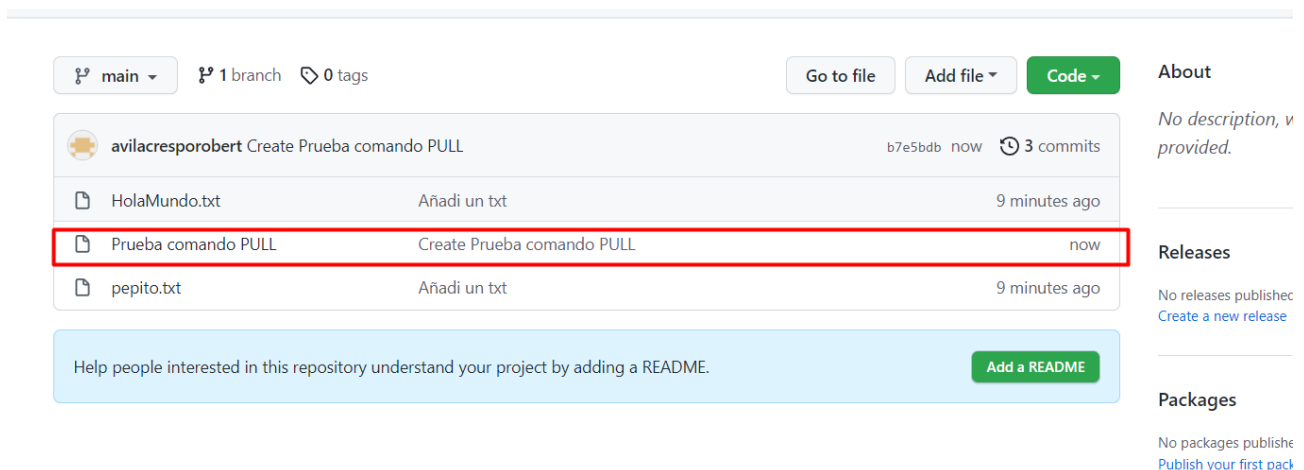
```
1
2
3 Que pasa compadre?
```

© 2021 GitHub, Inc. Terms Privacy Security Status Docs

COMANDO PULL:

El comando PULL sería todo lo contrario al push, lo que hace es algo que modificamos en github nos lo baja también al repositorio local. Vamos a realizar la misma prueba que antes pero a la inversa:

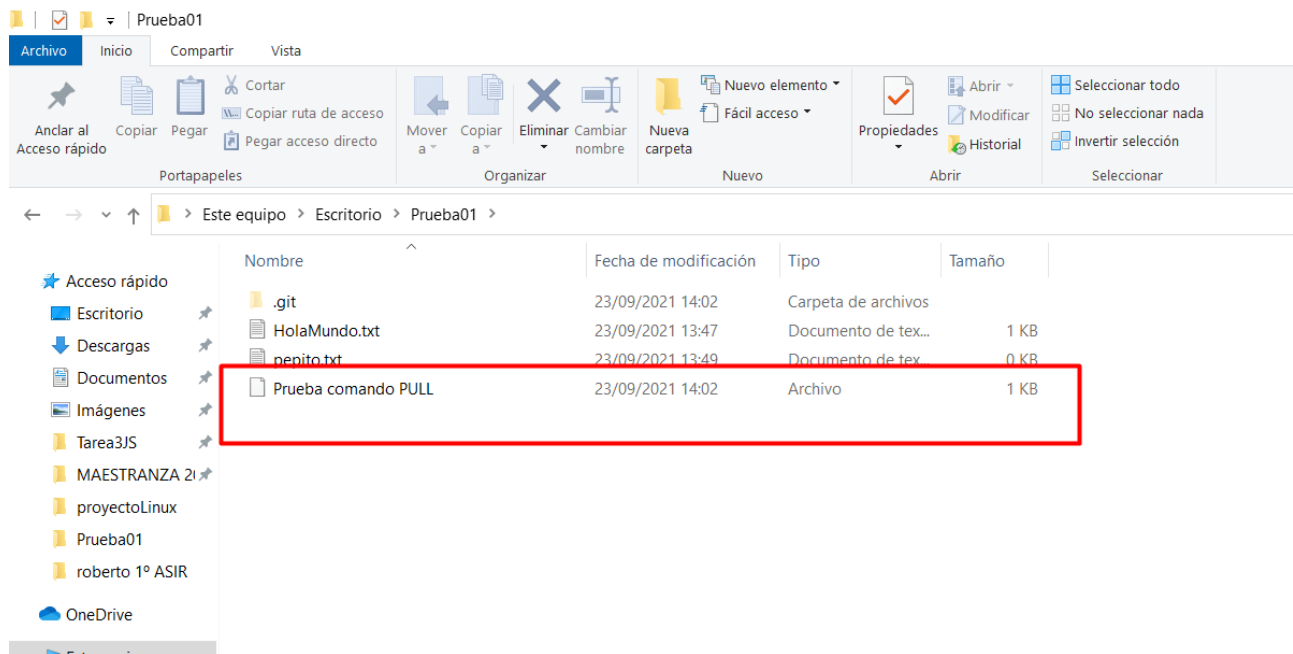
Creamos un nuevo archivo en nuestro repositorio github llamado:
Prueba comando PULL:



Realizamos el comando en la powershell, para comprobar si funciona:

```
PS C:\Users\avil\OneDrive\Escritorio\Prueba01> git pull https://github.com/avilacresporobert/Prueba001/
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 724 bytes | 90.00 KiB/s, done.
From https://github.com/avilacresporobert/Prueba001
* branch      HEAD      -> FETCH_HEAD
Updating 2e499db..b7e5bdb
Fast-forward
 Prueba comando PULL | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 Prueba comando PULL
PS C:\Users\avil\OneDrive\Escritorio\Prueba01>
```

Para terminar la comprobación, vamos al repositorio local y efectivamente vemos como se ha creado el nuevo archivo:



COMANDO CLONE

El comando clone lo que hace es descargar tu repositorio en el ordenador dandole la ruta y te actualiza directamente el repositorio:

Creamos un repositorio donde volcaremos el clonado:

```
Windows PowerShell
PS C:\Users\aaivil\OneDrive\Escritorio> mkdir Clonado

Directorio: C:\Users\aaivil\OneDrive\Escritorio

Mode                LastWriteTime         Length Name
----                -
d-----          23/09/2021   14:08                Clonado

PS C:\Users\aaivil\OneDrive\Escritorio> █
```

Realizamos el comando Clone, como muestro en la captura:

```
PS C:\Users\aaivil\OneDrive\Escritorio> git clone https://github.com/avilacresporobert/Prueba001/
Cloning into 'Prueba001'...
remote: Enumerating objects: 9, done.
remote: Counting objects: 100% (9/9), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 9 (delta 0), reused 6 (delta 0), pack-reused 0
Receiving objects: 100% (9/9), done.
PS C:\Users\aaivil\OneDrive\Escritorio> █
```

Y para la comprobación miramos que hemos clonado en dicha carpeta el repositorio github:

