Git & GITHUB

Marina Laguna Valdepeñas 2º DAW

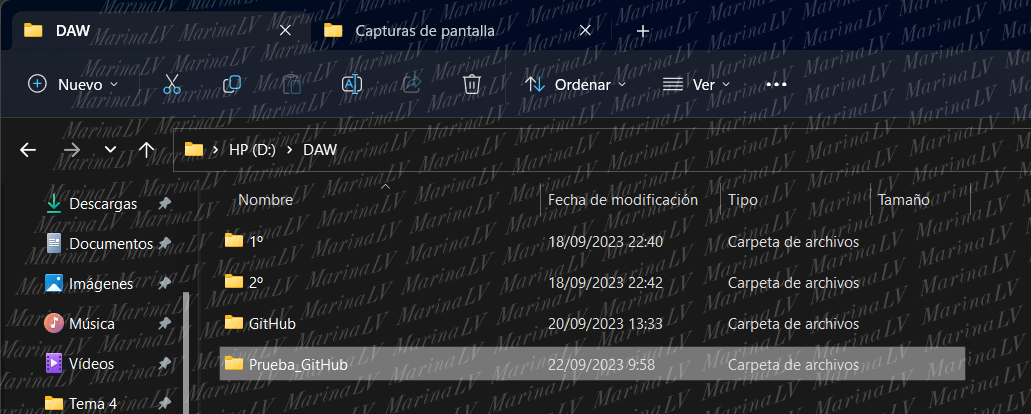
# GIT

## Repositorio Local

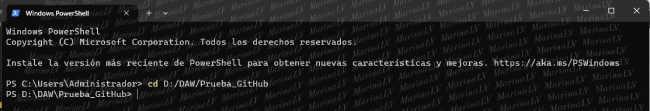
Lo primero que debemos saber es que un repositorio es un almacenamiento virtual donde guardaremos nuestro proyecto. El repositorio nos permite guardar versiones del código a las que podremos acceder cuando lo necesitemos.

Para crear un repositorio, tendremos que utilizar el comando git init al utilizar este comando se creará un subdirectorio en el directorio de trabajo actual y a su vez se creará una nueva rama inicial.

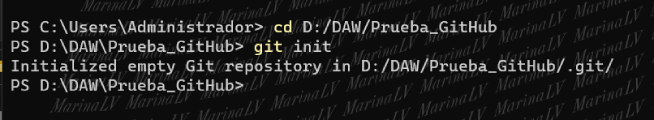
Primero crearemos la carpeta en la ubicación que deseemos, en mi caso la llamare Prueba\_GitHub



Ahora nos iremos a nuestra terminal (en mi caso uso la consola de Powershell), en la terminal lo primero que tenemos que hacer es ubicarnos donde hemos creado esa carpeta.

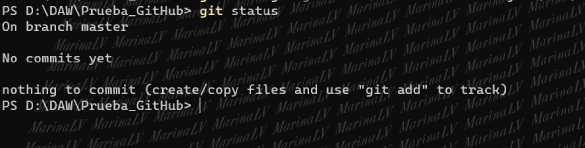


Para crear el repositorio local solo tendremos que ejecutar el comando git init, observaremos que al ejecutar este comando nos saldrá un mensaje que nos dirá que se ha creado el repositorio.



## Ver Estado del Repositorio

Si necesitamos ver el estado en el que se encuentra nuestro repositorio utilizaremos el comando git status, este comando nos mostrará el estado en el que se encuentra nuestro repositorio.

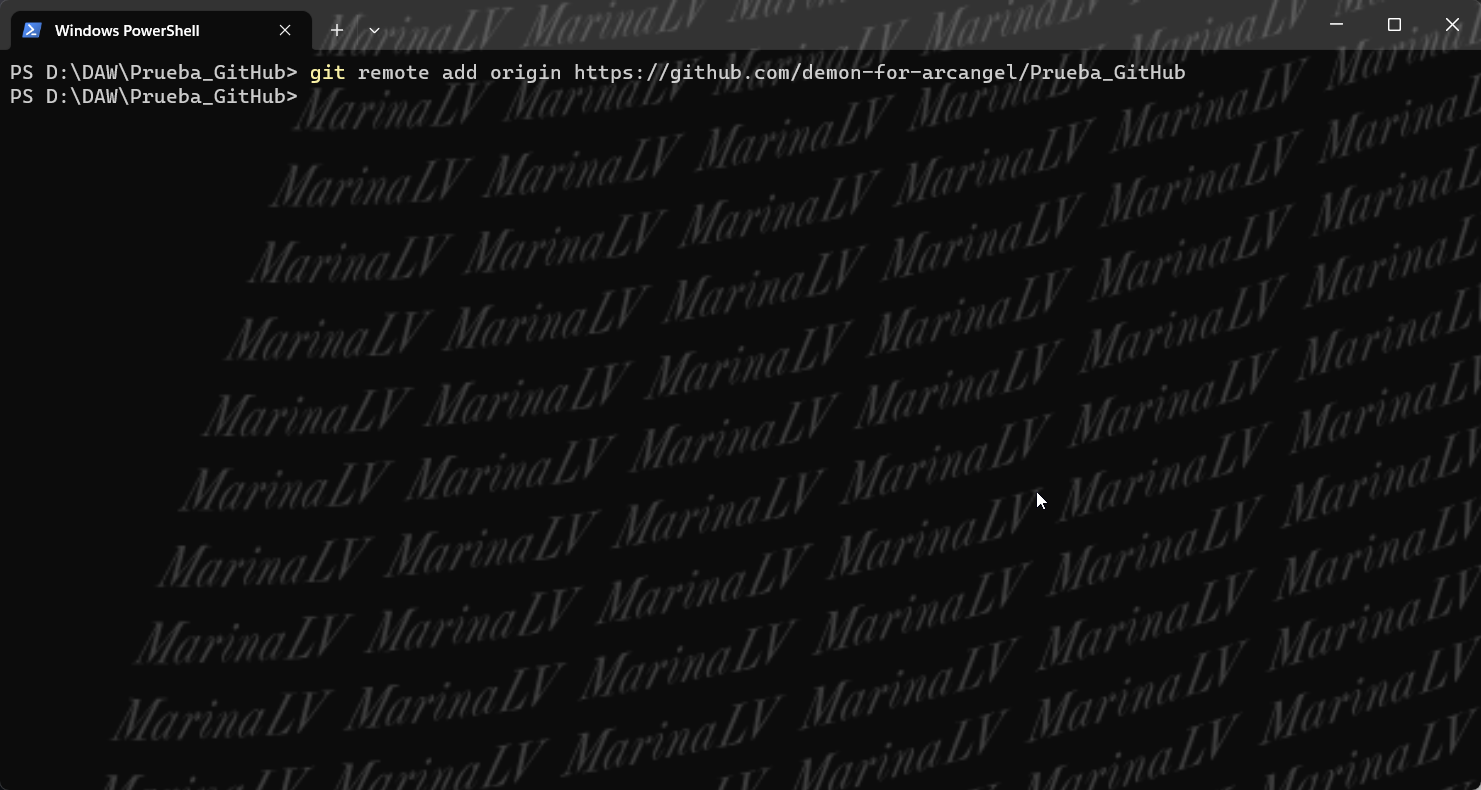


Como podemos observar nos pone la rama arriba en este caso es la rama Master (la principal) y debajo de la rama nos pone que no hay ningún commits echo.

## Enlazar el Repositorio Local de Git con GitHub

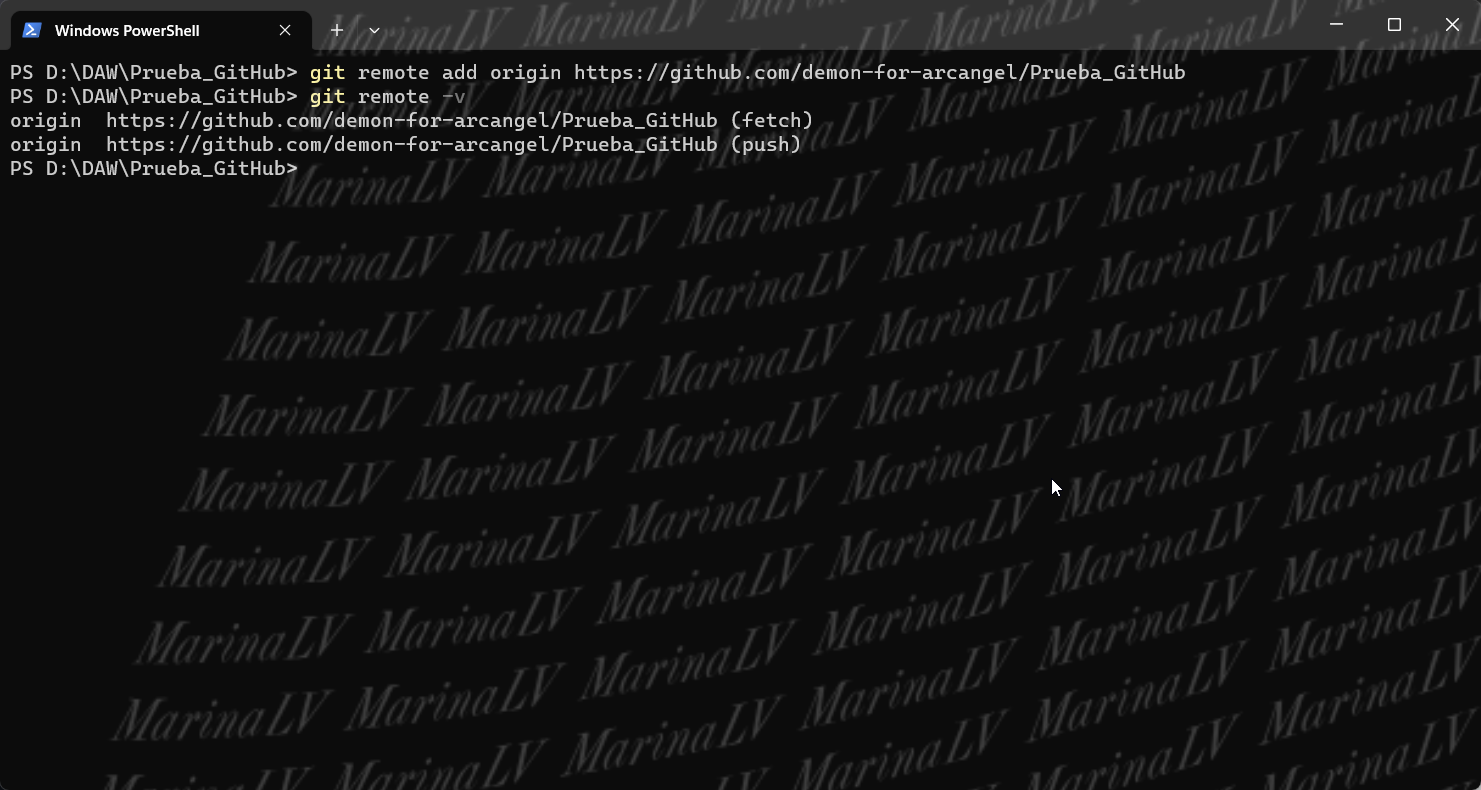
Para poder enlazar nuestro repositorio local con GitHub, necesitaremos saber la url de nuestro repositorio remoto (es el repositorio ubicado en GitHub). Utilizaremos el siguiente comando para realizar el enlace de los dos repositorios.

git remote add origin URL\_DEL\_REPOSITORIO\_GITHUB



Si necesitamos verificar el enlace utilizaremos el siguiente comando

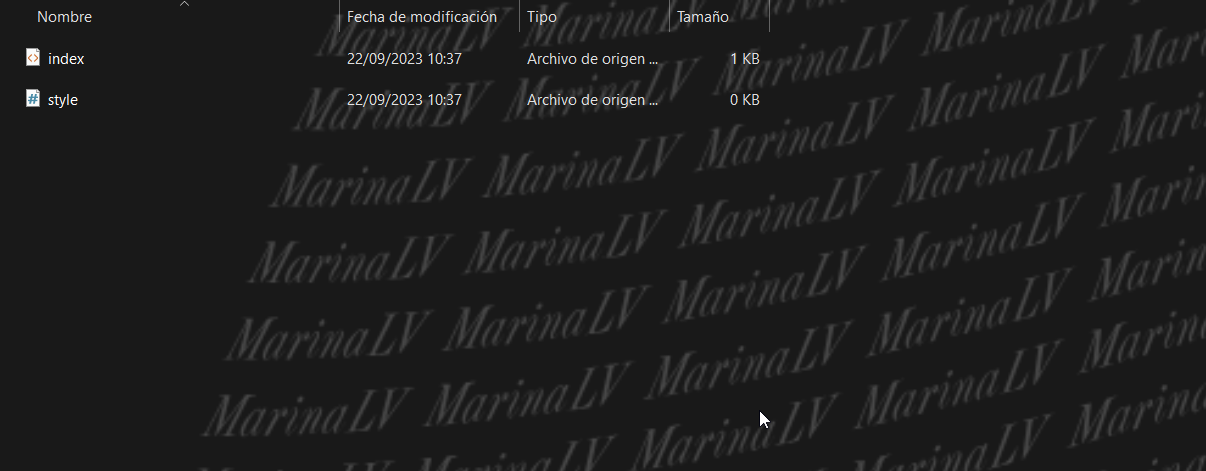
git remote -v



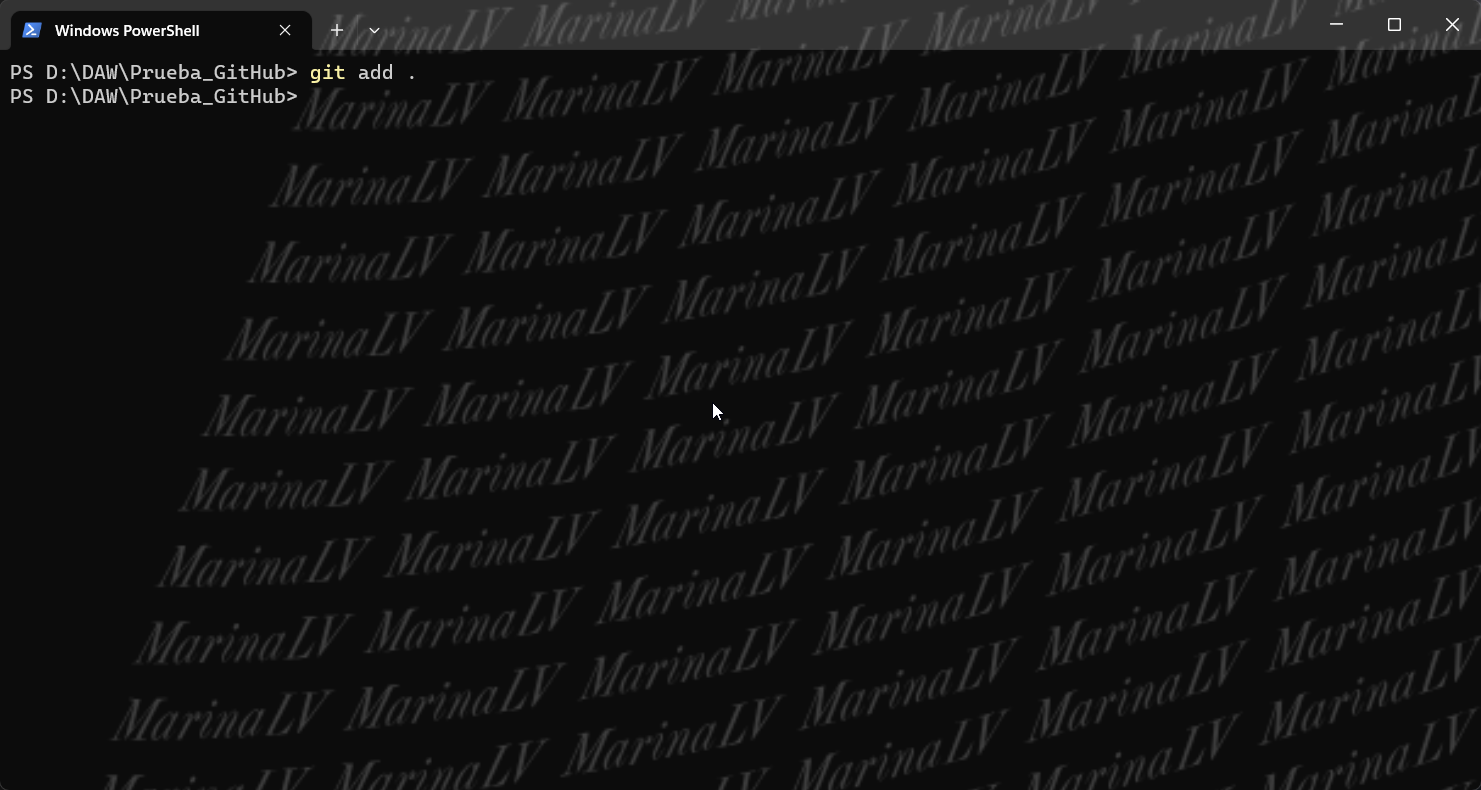
## Subida de Archivos

Para la subida de los archivos, tendremos que utilizar el comando git add.

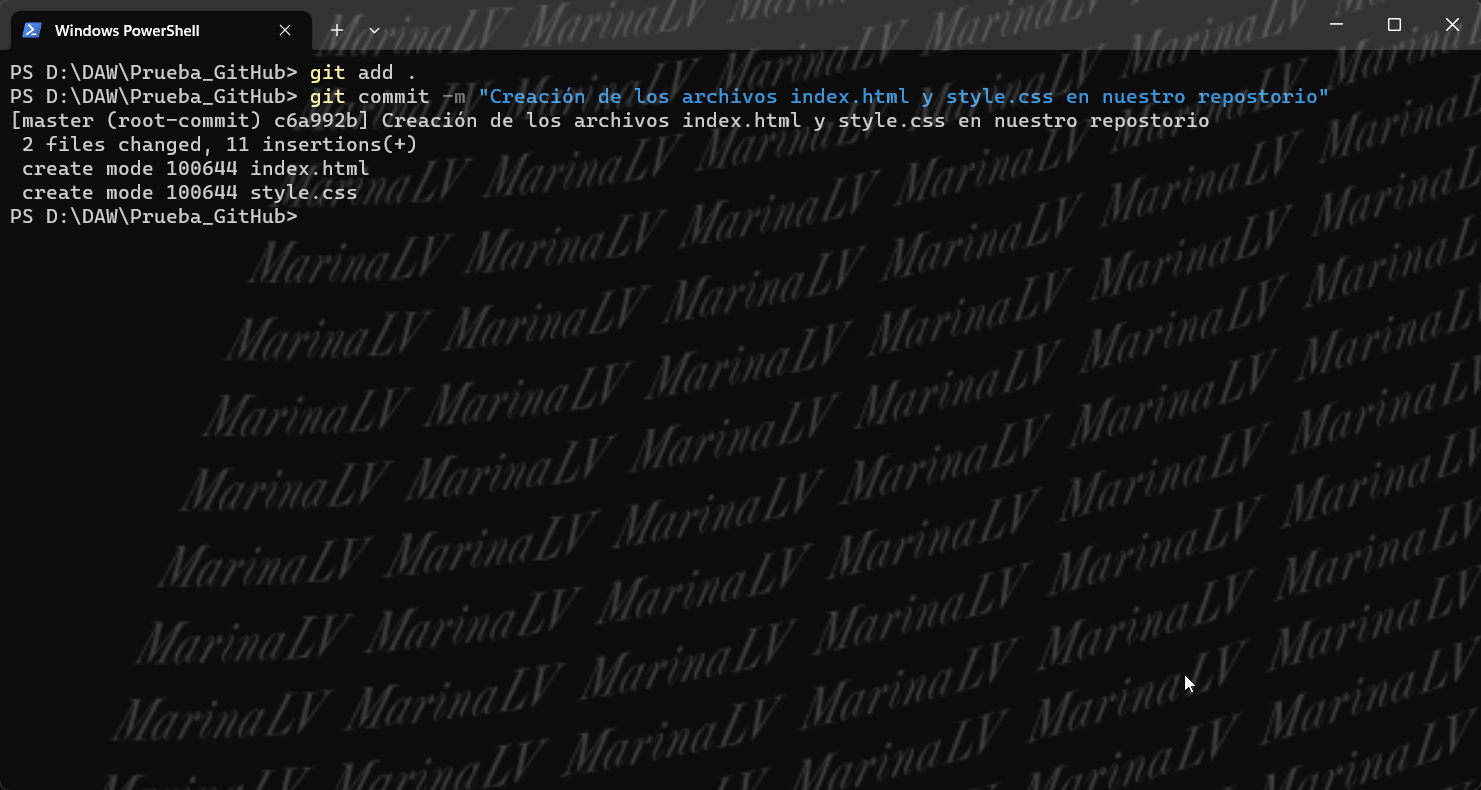
Para poder subir los archivos lo primero es añadir los archivos a nuestro repositorio local y después podremos utilizar nuestro comando en la terminal.



Si necesitamos subir muchos archivos utilizaremos git add .



Una vez ejecutado el comando git add realizaremos un commit para guardar los cambios en nuestro repositorio local. El commit es como una instantánea de los cambios que realizamos en un momento específico. El comando será el siguiente: git commit -m “Mensaje descriptivo del commit”



## Copia de Seguridad

Podemos realizar copias de seguridad de varias formas clonando el repositorio, crear una nueva rama que sea la copia de la principal, en mi caso explicare clonando el repositorio.

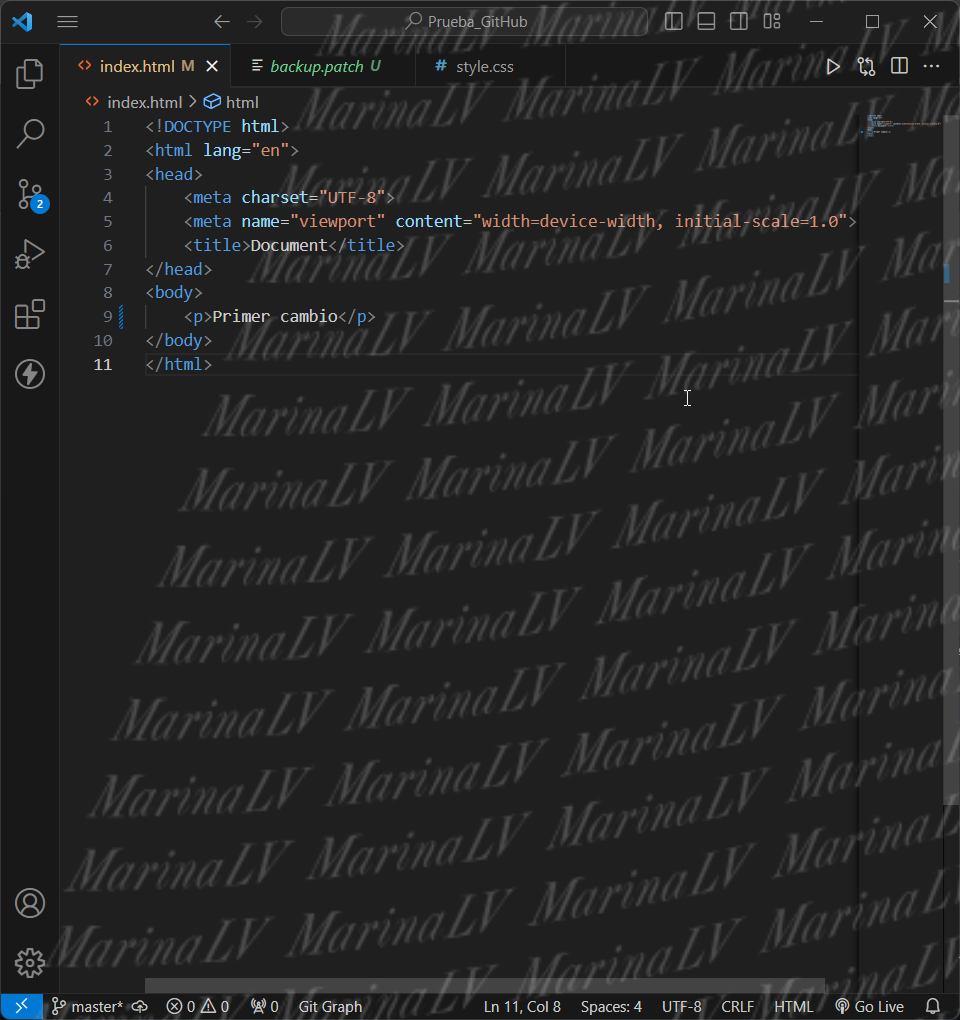
Si deseamos crear una copia de seguridad clonaremos nuestro repositorio actual en una nueva ubicación, para esto usaremos el siguiente comando

git clone ruta-del-repositorio ruta-de-destino



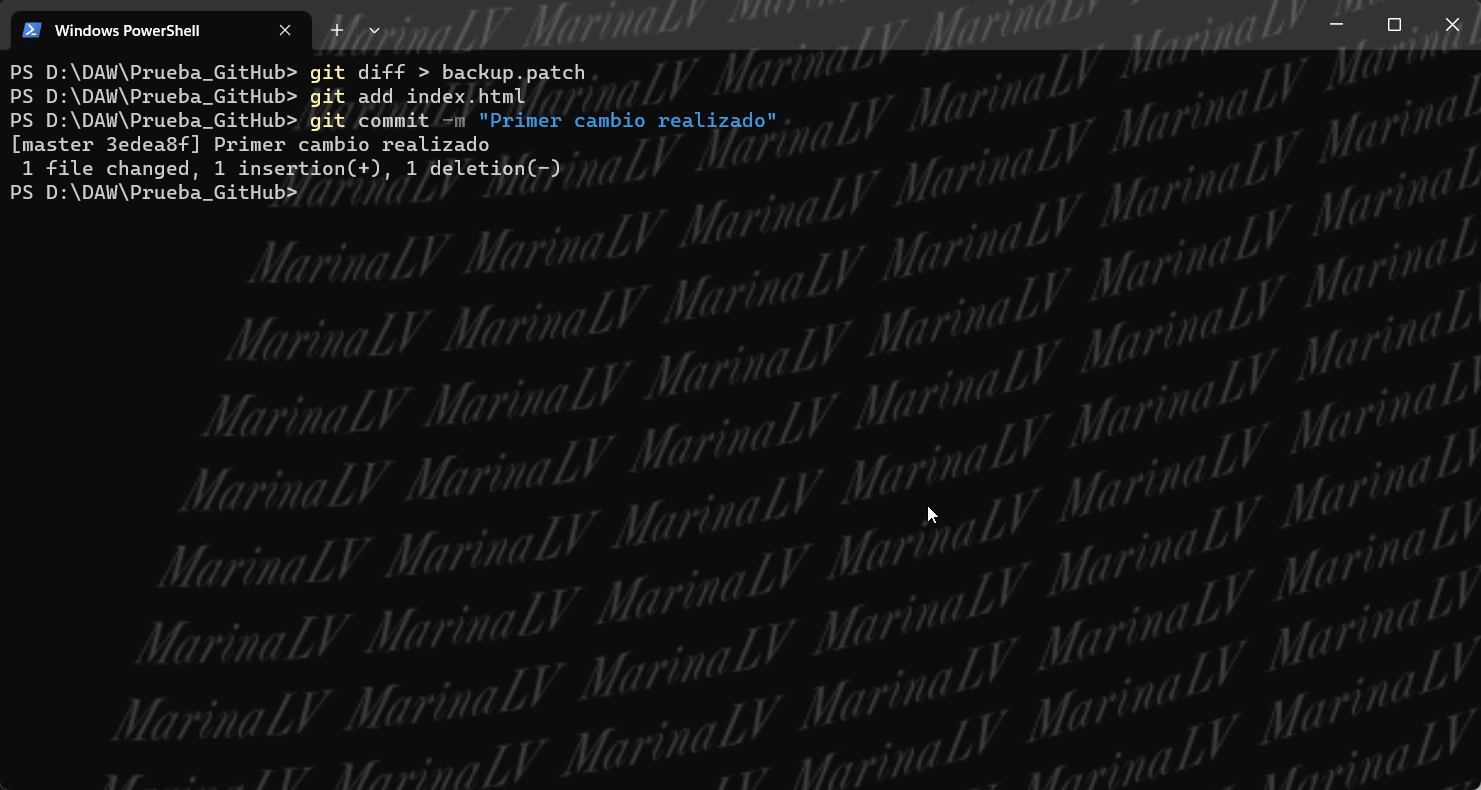
## Modificación de archivos

Para la modificación de los archivos, les haremos los cambios como le haríamos a un archivo normal y cuando lo tengamos listo, tendremos que realizar los mismos pasos que cuando subimos los archivos.



Primero realizaremos un git add para añadir nuestros archivos y seguidamente realizaremos un commit con un mensaje descriptivo de los cambios realizados.





## Restauración de archivos

Podemos recuperar los archivos de la clonación realizada como copia de seguridad, podremos coger los archivos de la copia de seguridad y llevárnoslo a nuestro repositorio o también podríamos directamente trabajar desde la clonación.

## Comandos más utilizados

Los comandos más utilizados que no hemos mencionado anteriormente son los siguientes:

* git push 🡪 envía los commits locales a un repositorio remoto. (Utilizado más adelante)
* git pull 🡪 trae los cambios desde un repositorio remoto y los fusiona con tu rama local.
* git status 🡪 muestra el estado actual del repositorio, incluyendo los archivos modificados y los que están en el área de preparación.
* git branch 🡪 lista, crea o elimina ramas. También lo podemos utilizar para cambiar entre ramas.
* git merge 🡪 fusiona los cambios de una rama en otra.
* git reset 🡪 deshace los cambios realizados en el área de preparación.

# GitHub

## Configuración de la Página de Presentación

Para la configuración de la Página de Presentación de nuestro GitHub necesitaremos crear un README.md.

Para que GitHub muestre el README.md al inicio de nuestro perfil deberemos cumplir unos prerrequisitos estos son los siguientes:

* Debemos crear un repositorio con un nombre que empate con el nombre de usuario que tenemos en GitHub.
* El repositorio debe ser público.
* En el repositorio deberá encontrarse el archivo de nombre README.md en su raíz.
* El archivo README.md contendrá cualquier tipo de contenido.

## Subir el Repositorio Local que tenemos en Git

Para conectar el repositorio local que tenemos ubicado en Git, tendremos que tener en nuestra cuenta de GitHub un repositorio para enlazarlos.

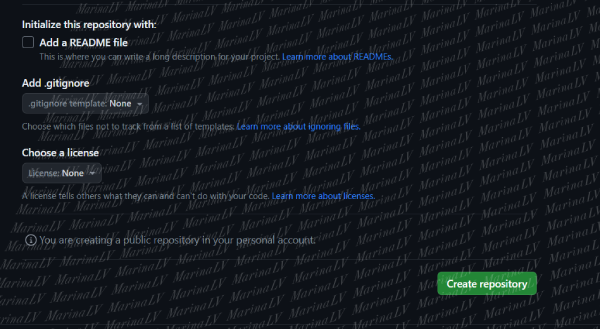
Para crear un repositorio en GitHub seguiremos los siguientes pasos:

Una vez dentro de la página de GitHub y hayamos iniciado sesión nos iremos al apartado de los Repositorios, en este apartado podremos ver en la esquina superior derecha un botón “New”, seleccionaremos este botón para crear un repositorio.



Veremos una página nueva en la que podremos configurar nuestro repositorio, elegir su nombre, ponerle una descripción, elegiremos si lo queremos público o privado, si queremos crear el README.md, una vez tengamos eso se creara nuestro repositorio seleccionando el botón “Create repository”.

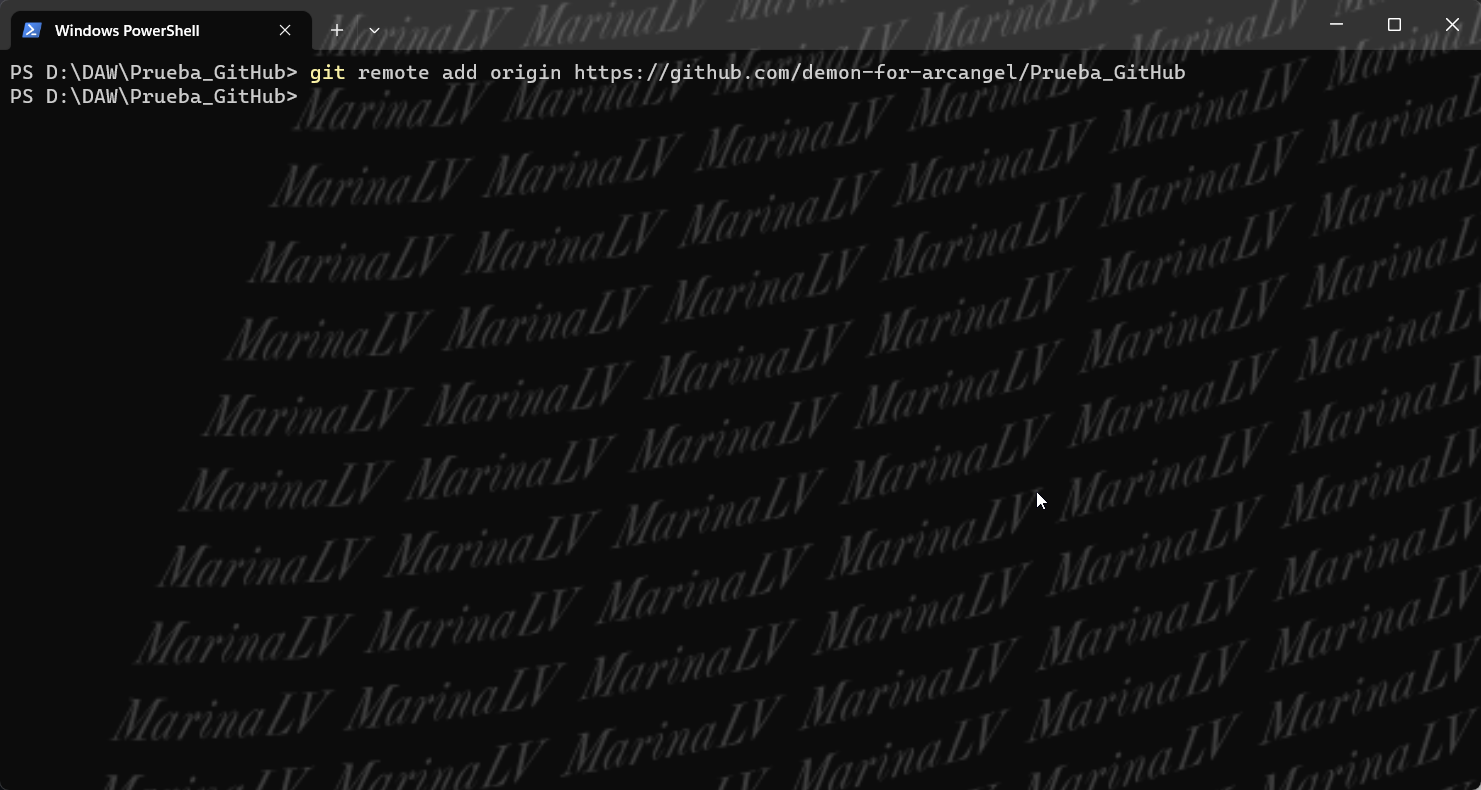




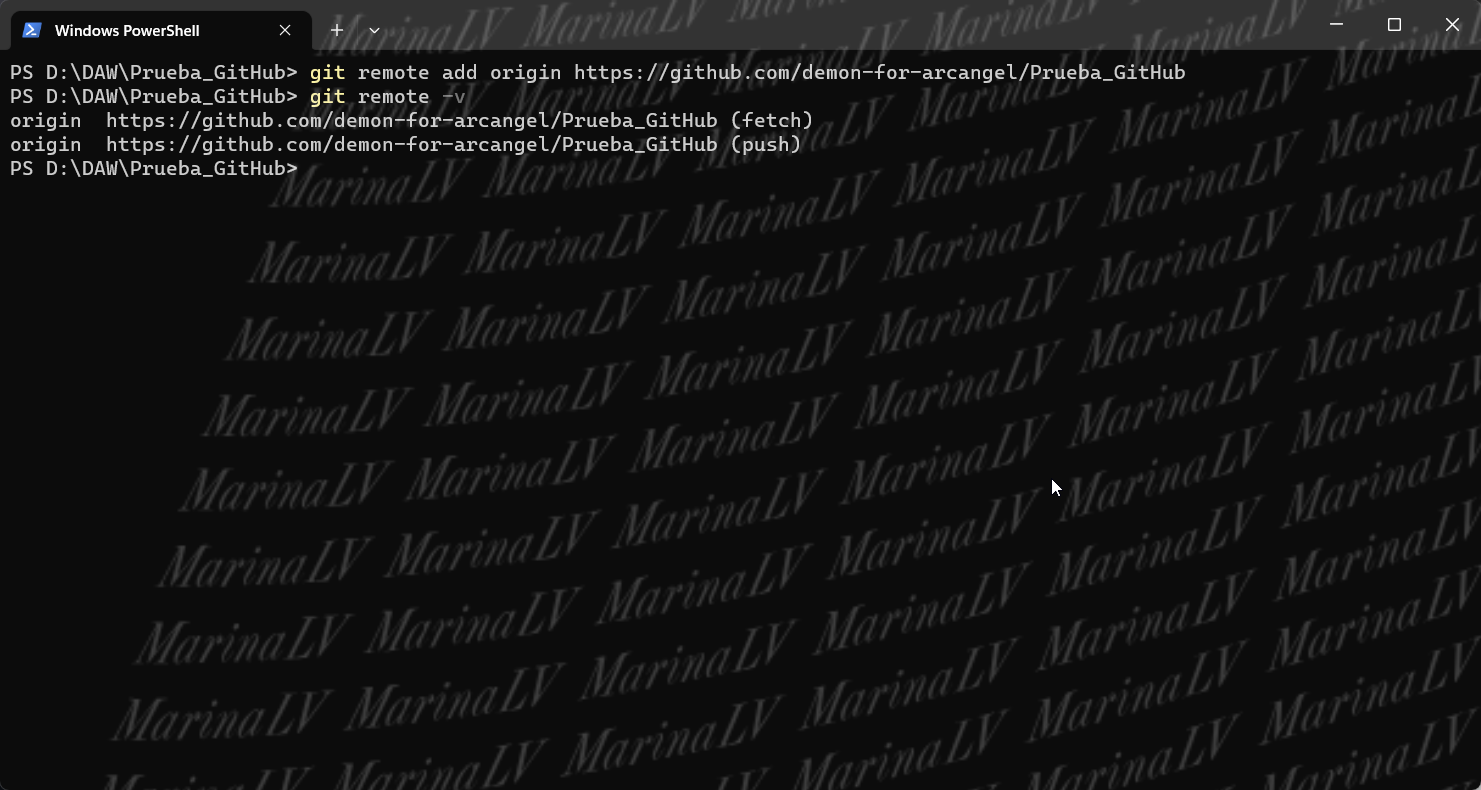
Nuestro repositorio creado quedará de la siguiente manera



Para poder subir nuestros archivos en GitHub nos faltará hacer el enlace de Git y GitHub, esto lo haremos desde la terminal (ubicados en nuestro repositorio) con el siguiente comando 🡪 git remote add origin URL\_REPOSITORIO\_REMOTO



Si queremos verificar la conexión ejecutaremos el comando 🡪 git remote -v



Con esto ya podremos subir nuestros archivos, para subir los archivos ejecutaremos los comandos git add para añadir los archivos, los commits describiendo los archivos que hemos subido y por último utilizaremos el comando git push -u origin RAMA para publicarlo en la rama de nuestro GitHub (si queremos subirlo a la rama principal utilizaremos git push -u origin master, esto lo veremos más adelante).

## Modificación de Archivos en Remoto

Para la modificación de los archivos en remoto lo realizaremos desde la página de GitHub, el archivo que queramos modificar en nuestra pantalla podremos ver una zona para ver el código pero sin poder editarlo para eso tendremos un botón en la esquina superior derecha que nos permitirá editar nuestro código.

Al terminar de editar el código podremos ver que tendremos un botón en la esquina superior derecha que podremos realizar el commit explicando los cambios que hemos realizado.

TERMINAR ESTA PARTE

## Actualización de Repositorios

## Eliminación y Recuperación del Proyecto

# GIT y GitHub

## Integración Git y GitHub en Visual Studio Codeç

La integración de Git y GitHub en VSC es sencilla debido a que ya nos dan la integración incluida, pero aún así lo podemos acompañar con algunas extensiones.

## Extensiones en Visual Studio Code

Una de las extensiones que tengo en Visual Studio Code es Git Graph

# GitKraquen