

GIT HUB

Creo una carpeta, con archivos dentro u carpetas con contenido, si están vacías no las podré iniciar

git add nombre agrego ese archivo

git add . agrego todos los archivos de ahí para abajo

git commit -m "agrego el nombre del commit"

git push origin main subo los archivos a mi repositorio remoto

git pull origin main descargo los archivos desde mi repositorio remoto (sólo los que no tengo o fueron modificados)

git remote user.name "Mario" vinculo mi nombre de Usuario (tiene que ser el de GIT HUB)

git remote --global user.name "Mario" lo vinculo de ahí en mas

git remote user.email "marioandres.....@gmail.com" vinculo mi email (tiene que ser el de GIT HUB)

git remote --global user.email "marioandres.....@gmail.com" lo vinculo de ahí en mas

git pull origin main + dirección que me brinda el GIT HUB del repositorio
(<https://github.com/Mario/nombre> de la carpeta)

git remote -v pruebo si la carpeta está sincronizada con GIT HUB

git status me muestra el estado de los archivos

git log me muestra los commits (en mi máquina)

git log --oneline me muestra los commits con poco detalle (en mi máquina)

git clone <https://github.com/Mario/nombre> crea una copia exacta de los archivos en la nube,
descarga los archivos si no los tuviéramos, si vamos a trabajar en una pc nueva por ej. No es
necesario sincronizarla nuevamente

SI NOS ENCONTRAMOS MODIFICADO EL REPOSITORIO!!!!!!

Primero nos traemos los archivos

git pull origin main

resolvemos las diferencias y lo subimos

git add nombre - git commit -m "agrego el nombre del commit" - git pull origin main

NUEVA RAMA

git branch "nombre de la rama"

git branch me muestra las ramas

git checkout para cambiarme a esa rama

Comandos paso a paso para crear un repositorio local

```
>_ git init // Crea el repositorio
```

```
>_ git config user.name "nombreUsuario" // Agrega nuestra identidad
```

```
>_ git config user.email "emailUsuario" // Agrega nuestro e-mail
```

```
>_ git remote add origin http://... // Apunta al repositorio remoto
```

Comandos paso a paso para subir cambios

```
>_ git add . // Agrega todos los archivos
```

```
>_ git commit -m "mensaje" // Commitea los cambios hechos
```

```
>_ git push origin master // Envía los cambios al repositorio remoto
```

```
>_ git status // Realiza un seguimiento de los estados de los archivos
```

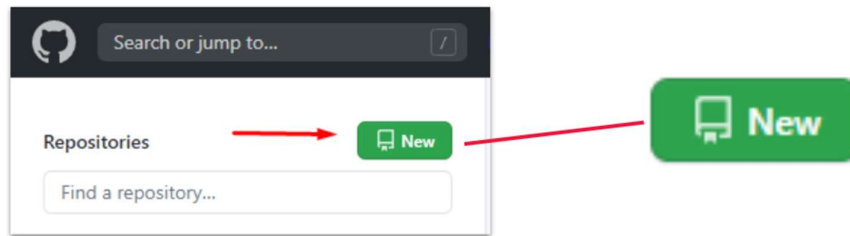
otros comandos

```
>_ git status // Realiza un seguimiento de los estados de los archivos
```

```
>_ git pull// Descarga los cambios que existen en el repositorio remoto
```

Paso a paso para crear un repositorio desde github

- 1 En nuestra cuenta de GitHub hacemos clic en New.



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 *

Great repository names are short and memorable. Need inspiration? [How do I choose a name?](#)

Description (optional)

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can use 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](#)

2

Elegimos un nombre para el repositorio y hacemos clic en Create repository.

Repository name *

/

3

Con el link que nos provee GitHub de nuestro repositorio, ingresamos el siguiente comando:

```
>_ git clone linkDelRepo // Clonamos el repositorio en nuestra máquina
```

Cómo aceptar invitación al repositorio

Si estamos trabajando en equipo y otro integrante nos invita a colaborar en su repositorio, lo que tenemos que hacer es lo siguiente:

- 1 Revisar nuestro correo y hacer clic donde dice [accept or decline](#).
- 2 Nos redireccionará a GitHub donde finalmente aceptaremos la invitación a colaborar.
- 3 Una vez aceptada la invitación con git clone, estamos listos para trabajar como colaboradores del repositorio.

