

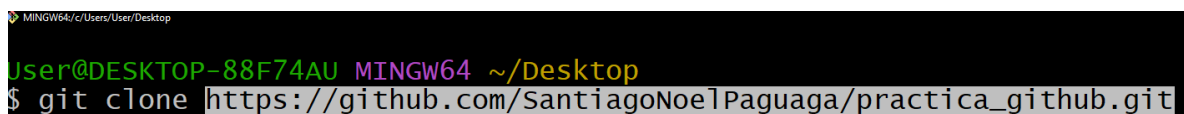
Primer Paso:

Posicionarse en cualquier parte del escritorio y dar click derecho con el mouse, seleccionamos la opción git bash here para abrir la terminal.

Segundo Paso:

Una vez estamos en la terminal debemos clonar el repositorio(les dejo el link por las dudas:

https://github.com/SantiagoNoelPaguaga/practica_github.git), para ello escribimos lo siguiente:

A screenshot of a terminal window with a black background. The title bar at the top reads 'MINGW64/c/Users/User/Desktop'. The prompt is 'User@DESKTOP-88F74AU MINGW64 ~/Desktop'. The command '\$ git clone https://github.com/SantiagoNoelPaguaga/practica_github.git' is entered and highlighted with a light gray background.

```
MINGW64/c/Users/User/Desktop
User@DESKTOP-88F74AU MINGW64 ~/Desktop
$ git clone https://github.com/SantiagoNoelPaguaga/practica_github.git
```

Nos mostrará en pantalla algo así:

```
MINGW64/c/Users/User/Desktop
User@DESKTOP-88F74AU MINGW64 ~/Desktop
$ git clone https://github.com/SantiagoNoelPaguaga/practica_github.git
Cloning into 'practica_github'...
remote: Enumerating objects: 57, done.
remote: Counting objects: 100% (57/57), done.
remote: Compressing objects: 100% (25/25), done.
remote: Total 57 (delta 3), reused 54 (delta 0), pack-reused 0
Receiving objects: 100% (57/57), 5.47 KiB | 2.74 MiB/s, done.
Resolving deltas: 100% (3/3), done.

User@DESKTOP-88F74AU MINGW64 ~/Desktop
$
```

Tercer Paso: debemos movernos a la carpeta que acabamos de clonar, para ello tenemos dos opciones, vamos directamente al escritorio donde se nos generó la carpeta y le damos click derecho sobre ella, y seleccionamos git bash here, o bien nos movemos desde la consola de bash que tenemos abierta de la siguiente manera:

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop
$ cd practica_github/

User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (main)
$
```

Podremos ver ahora a la derecha la rama en la que nos encontramos dentro de los paréntesis (main).

Cuarto Paso:

Una vez que estamos posicionados en la rama main, vamos a crear una nueva rama, para ello escribimos lo siguiente:

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (main)
$ git checkout -b elNombreQueDeseen
Switched to a new branch 'elNombreQueDeseen'

User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (elNombreQueDeseen)
$
```

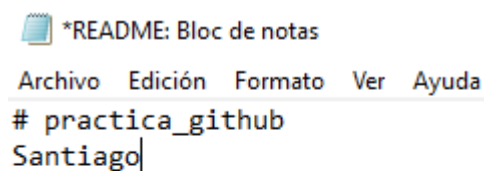
Dicho comando no solo nos crea la nueva rama, también nos mueve hacia la rama que acabamos de crear, como podemos ver en los paréntesis a la derecha, ya no nos encontramos en main.

Quinto Paso:

Ahora dentro de la rama que acabamos de crear vamos a modificar el archivo README.md para ellos colocamos lo siguiente:

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (e1NombreQueDeseen)
$ notepad README.md |
```

Al escribir dicho comando y presionar enter se nos abrirá el blog de notas, al mismo debemos modificarlo agregando nuestro nombre en la segunda línea, debería quedar algo así:



```
*README: Bloc de notas
Archivo  Edición  Formato  Ver  Ayuda
# practica_github
Santiago|
```

Le damos a Archivo y guardamos los cambios, y podemos cerrar el blog de notas.

Sexto Paso:

Nuevamente en la consola ahora deberemos agregar el archivo que

acabamos de modificar, para ello ingresamos:

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (elNombreQueDeseen)
$ git add README.md
```

Luego realizamos el correspondiente commit:

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (elNombreQueDeseen)
$ git commit -m "El mensaje que quieran"
[elNombreQueDeseen 417211a] El mensaje que quieran
1 file changed, 1 insertion(+), 15 deletions(-)

User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (elNombreQueDeseen)
$ |
```

Y Por último haremos el push, pero siempre dentro de nuestra rama:

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (elNombreQueDeseen)
$ git push origin elNombreQueDeseen
```

Ya tenemos nuestro README.md modificado con nuestro nombre y con su correspondiente push en nuestra rama.

Séptimo Paso:

Ahora debemos fusionar nuestra rama con la rama main, para ello, es importante primero moverse nuevamente a la rama main de la siguiente manera:

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (elNombreQueDeseen)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (main)
$ |
```

Notar nuevamente que en los paréntesis a la derecha volvemos a estar dentro de la rama main.

Para realizar la fusión de las ramas escribimos lo siguiente.

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (main)
$ git merge elNombreQueDeseen
Updating 755b85a..417211a
Fast-forward
 README.md | 16 +-----
 1 file changed, 1 insertion(+), 15 deletions(-)

User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (main)
$
```

Octavo Paso:

Por último debemos hacer el push a nuestro servidor remoto, pero puede ocurrir un conflicto llegado a este paso, al momento de clonar nuestro repositorio, nuestro archivo README.md solo contenía la primera línea, si otros usuarios agregaron algo después de que nosotros clonamos el repositorio, podemos recibir un mensaje de error, para ello antes podemos intentar hacer un pull(actualiza los datos):

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (main)  
$ git pull origin main
```

Y luego realizamos el push:

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (main)  
$ git push origin main|
```

Noveno Paso:

Si por algún motivo el conflicto persiste y nos aparece lo siguiente:

```
User@DESKTOP-88F74AU MINGW64 ~/practica_github (main|MERGING)
$ git push origin main
fatal: credential-cache unavailable; no unix socket support
To https://github.com/SantiagoNoelPaguaga/practica_github.git
 ! [rejected]        main -> main (non-fast-forward)
error: failed to push some refs to 'https://github.com/SantiagoNoelPaguaga/practica_github.git'
hint: updates were rejected because the tip of your current branch is behind
hint: its remote counterpart. Integrate the remote changes (e.g.
hint: 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

User@DESKTOP-88F74AU MINGW64 ~/practica_github (main|MERGING)
$
```

Volveremos a intentar hacer el pull:

```
User@DESKTOP-88F74AU MINGW64 ~/practica_github (main|MERGING)
$ git pull
error: Pulling is not possible because you have unmerged files.
hint: Fix them up in the work tree, and then use 'git add/rm <file>'
hint: as appropriate to mark resolution and make a commit.
fatal: Exiting because of an unresolved conflict.

User@DESKTOP-88F74AU MINGW64 ~/practica_github (main|MERGING)
$
```

De esta manera conseguimos que nos genere el conflicto.

Ahora el siguiente paso es abrir nuevamente nuestro Readme.md con el blog de notas escribiendo nuevamente:

```
User@DESKTOP-88F74AU MINGW64 ~/Desktop/practica_github (e1NombreQueDeseen)
$ notepad README.md |
```

Pero ahora nuestro blog de notas se verá algo así:

README: Bloc de notas

Archivo Edición Formato Ver Ayuda

practica_github

Santiago Paguaga

<<<<<< HEAD



Mentor inExperto MarcossLug

Santiago Paguaga

Prueba

=====

Prueba 2

>>>>>> Santiago



Debemos borrar las 3 líneas que están subrayadas en rojo y nos quedaría algo así:

*README: Bloc de notas

Archivo Edición Formato Ver Ayuda

practica_github

Santiago Paguaga

Mentor inExperto MarcossLug

Santiago Paguaga

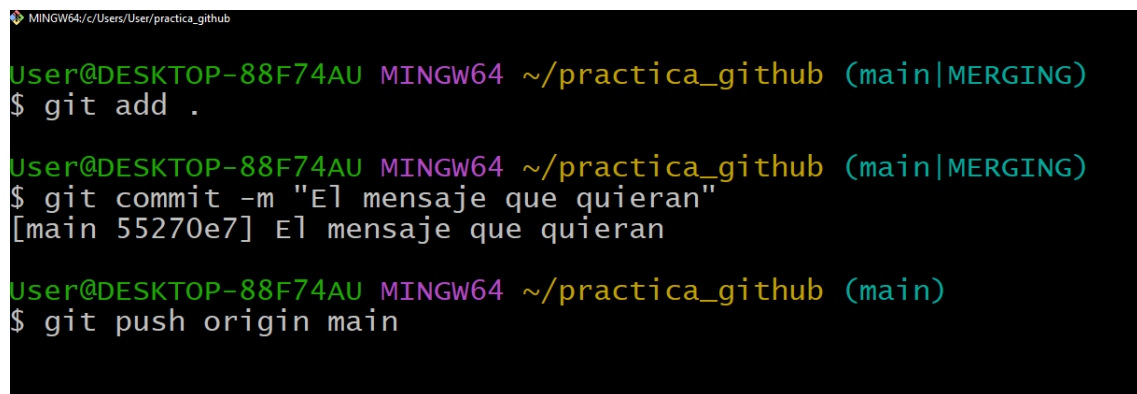
Prueba

Prueba 2

|

Es importante solo borrar las líneas pero no quitar los espacios que aparecen entre las demás palabras, luego le damos a guardar los cambios y cerramos el blog de notas.

Por último ya podemos agregar nuestro archivo README.md mediante el comando add, luego realizar el commit y por último subirlo a nuestro servidor remoto con push de la siguiente manera:



```
MINGW64/c/Users/User/practica_github
User@DESKTOP-88F74AU MINGW64 ~/practica_github (main|MERGING)
$ git add .
User@DESKTOP-88F74AU MINGW64 ~/practica_github (main|MERGING)
$ git commit -m "El mensaje que quieran"
[main 55270e7] El mensaje que quieran
User@DESKTOP-88F74AU MINGW64 ~/practica_github (main)
$ git push origin main
```

Y eso ya sería todo, deberían poder ver su nombre en el repositorio remoto junto al de los demás integrantes.

Aclaración: si el ejercicio lo vuelven a comenzar desde ahora el README.md al clonar el repositorio ya tiene varios nombres agregados que son los que fuimos haciendo el push durante la clase.