

Trabajo Práctico 2

1)

GitHub es una comunidad donde podemos compartir nuestros repositorios de manera pública o privada.

Para crear un repositorio en GitHub, primero debo crear una cuenta. Ya en el perfil, voy a tener un listado de repositorios, si es que ya tengo creados. Sino, puedo crear uno haciendo click en el botón **New**. Ahí ya se va a poder crear el repositorio con su nombre, descripción, visibilidad (público o privado) archivo README (o no). Después de elegir sus características se selecciona **Create repository**.

Para crear una rama en Git, debo escribir el comando **git branch nombreRama**, donde nombreRama es el nombre de la nueva rama.

Para cambiar a una rama debo escribir **git checkout nombreRama**, donde nombreRama es el nombre de la rama a la que me quiero cambiar.

Para fusionar ramas en GitHub, debo pararme primero sobre la rama a la que se le van a aplicar los cambios de la otra rama y escribir el comando **git merge otraRama**, donde otraRama es el nombre de la rama que se va a fusionar.

Para crear un commit, debo ejecutar **git commit -m "mensaje del commit"**. Antes debo usar **git add** . para que git haga el seguimiento de mis archivos.

Para enviar un commit a GitHub, primero debo asociar el repositorio remoto al repositorio local con **git remote add origin linkRepo**, donde linkRepo es la URL de mi repositorio remoto de GitHub. Después, debo enviar los cambios con **git push -u origin master** (si master es el nombre de mi rama).

Un repositorio remoto es una copia de mi proyecto alojada en un servidor, es decir, en la nube.

Para tirar cambios de un repositorio remoto debo usar **git pull**.

Un fork es una copia de un repositorio creada en una cuenta diferente permitiendo desarrollar cambios, sin afectar el original.

Para crear un fork debo ir al repositorio que quiero copiar y selecciono **Fork**. Ahí puedo cambiar la descripción del fork si lo deseo y después debo seleccionar **Create fork**.

Para hacer un pull request, siendo colaborador, hay que seguir los siguientes pasos:

Clonar el repositorio de manera local con **git clone linkRepo**.

Crear una nueva rama para los cambios con **git branch nombreRama**

Después agrego y commiteo con **git add .** y **git commit -m “mensaje commit”**

Luego subo los cambios a la rama con **git push origin nombreRama.**

Para finalizar, voy al repositorio en GitHub y selecciono para abrir un pull request. Hago click en **Compare & pull request.** Agrego un título y descripción y selecciono **Create pull request.**

Para aceptar una solicitud de extracción, voy al repo en GitHub y me dirijo a la pestaña de **Pull Requests.** Reviso la solicitud de extracción, los conflictos si los hay y después selecciono **Merge pull request.** Puedo agregar un mensaje de fusión si lo creo necesario y después confirmo la fusión. También, puedo eliminar la rama si quiero.

Una etiqueta o tag es una referencia que apunta a un commit en específico en el historial del repositorio. Para crear una etiqueta debo ejecutar **git tag nombreEtiqueta.** Para enviar la etiqueta a GitHub, ejecuto **git push origin nombreEtiqueta.**

Un historial en git es un listado de todos los commits que se hicieron en un repositorio. Para ver el historial, debo ejecutar **git log.**

Hay distintas maneras de buscar en el historial de git:

- Por mensaje específico: **git log --grep="mensaje a buscar"**
- Por autor: **git log --author="nombre autor"**

Entre otras.

Para eliminar commits locales, puedo ejecutar **git reset --hard idCommit,** donde idCommit es el id del commit a eliminar.

Un repositorio privado en GitHub es un repositorio cuyo acceso está restringido y solo las personas que han sido invitadas pueden ver y colaborar en el repositorio.

Para crear un repositorio, voy a Inicio y selecciono el botón **+**. Ahí selecciono **New repository.**

En el formulario de creación, debo seleccionar la opción **Private.**

Para invitar a alguien a mi repo privado, debo ir a la pestaña **Settings** y seleccionar **Manage Access.** Ahí elijo **Invite a collaborator.** Luego ingreso el nombre de usuario o correo de la persona a invitar y elijo **Add.**

Un repositorio público es accesible para cualquier persona. Cualquier persona puede ver, clonar y contribuir al proyecto.

Para crear un repo público en GitHub, debo seguir los mismos pasos que para crear uno privado, pero seleccionando la opción **Public.**

Para compartir un repositorio público en GitHub, simplemente debo compartir su URL.

2)

```
MINGW64/c/Users/franc/Desktop/PRACTICA-GIT
franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT
$ git init
Initialized empty Git repository in C:/Users/franc/Desktop/PRACTICA-GIT/.git/

franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT (master)
$ git add .

franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT (master)
$ git commit -m "Agregando mi-archivo.txt"
[master (root-commit) 86dd83e] Agregando mi-archivo.txt
1 file changed, 1 insertion(+)
create mode 100644 mi-archivo.txt

franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT (master)
$ git remote add origin https://github.com/fleoneilherrera/repo-practica.git

franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 240 bytes | 240.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/fleoneilherrera/repo-practica.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT (master)
$ git branch franco

franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT (master)
$ git checkout franco
Switched to branch 'franco'

franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT (franco)
$ git add .

franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT (franco)
$ git commit -m "Agrego un README"
[franco 1d7d45d] Agrego un README
1 file changed, 1 insertion(+)
create mode 100644 README.md

franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT (franco)
$ git push -u origin franco
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 315 bytes | 315.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'franco' on GitHub by visiting:
remote:   https://github.com/fleoneilherrera/repo-practica/pull/new/franco
remote:
To https://github.com/fleoneilherrera/repo-practica.git
 * [new branch]      franco -> franco
branch 'franco' set up to track 'origin/franco'.

franc@Franco MINGW64 ~/Desktop/PRACTICA-GIT (franco)
$ |
```

3)

```
MINGW64:/c/Users/franc/Desktop/PRACTICA_conflict_exercise/conflict-exercise

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise
$ git clone https://github.com/fleone1herrera/conflict-exercise.git
Cloning into 'conflict-exercise'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise
$ cd conflict-exercise

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main)
$ git branch feature-branch

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main)
$ git checkout -b feature-branch
fatal: a branch named 'feature-branch' already exists

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main)
$ git checkout feature-branch
Switched to branch 'feature-branch'

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (feature-branch)
$ git add .

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (feature-branch)
$ git commit -m "Added a line in feature-branch"
[feature-branch 8c3ef38] Added a line in feature-branch
1 file changed, 1 insertion(+)

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (feature-branch)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main)
$ git add .

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main)
$ git commit -m "Added a line in main branch"
[main 9b28251] Added a line in main branch
1 file changed, 1 insertion(+)

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main)
$ git merge feature-branch
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main|MERGING)
$ git add .


franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main|MERGING)
$ git commit -m "Resolved merge conflict"
[main 8127be7] Resolved merge conflict
```


```
franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main)
$ git push origin main
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (9/9), 779 bytes | 779.00 KiB/s, done.
Total 9 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/fleone1herrera/conflict-exercise.git
91cd892..8127be7 main -> main


franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main)
$ git push origin feature-branch
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature-branch' on GitHub by visiting:
remote:   https://github.com/fleone1herrera/conflict-exercise/pull/new/feature-branch
remote:
To https://github.com/fleone1herrera/conflict-exercise.git
* [new branch]   feature-branch -> feature-branch

franc@Franco MINGW64 ~/Desktop/PRACTICA_conflict_exercise/conflict-exercise (main)
$ |
```

Commits


 main

 All users



 All time

Commits on Mar 26, 2025


Resolved merge conflict

 fleoneilherrera committed 2 minutes ago



8127be7


Added a line in main branch

 fleoneilherrera committed 5 minutes ago



9b28251


Added a line in feature-branch

 fleoneilherrera committed 5 minutes ago

8c3ef38

Initial commit

 fleoneilherrera authored 10 minutes ago

Verified91cd892

