

Trabajo práctico N° 2:

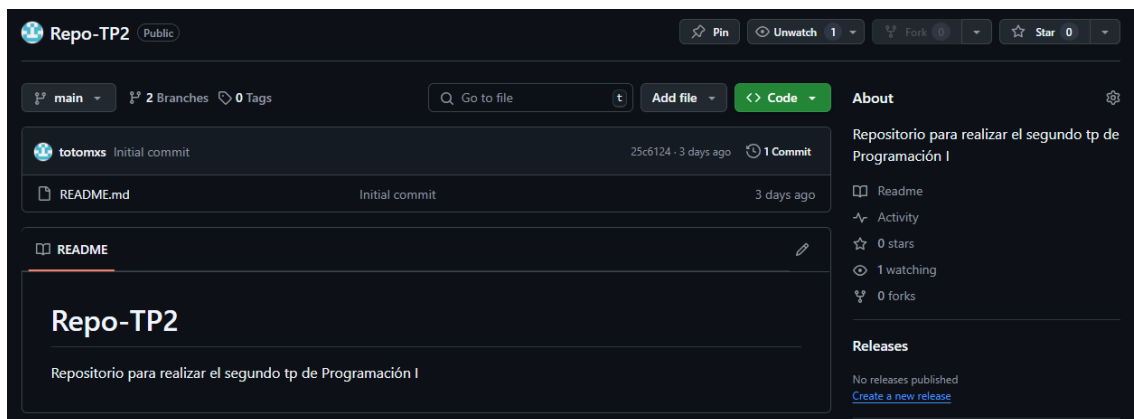
Trabajo colaborativo.

ACTIVIDAD 1.

1. GitHub es una comunidad donde se pueden compartir los repositorios de manera pública o privada. Tiene un formato similar al de una red social donde se puede “seguir” a otros usuarios y ver sus trabajos (si éstos son públicos).
2. Para crear un repositorio en GitHub lo primero que hay que hacer es crearse una cuenta en dicha página. Una vez en el perfil creado ir al botón con forma de signo mas (+) y clickear en “nuevo repositorio”. Ahí se configuran los datos pedidos y se le da click al botón de “crear repositorio”.
3. Para crear una rama en Git hay que escribir el comando “git branch *nombre*” en el editor de texto, donde *nombre* es la denominación que se le da a dicha rama.
4. Para cambiar a una rama en Git hay que escribir el comando “git checkout *nombre*”. Esto hará que pase de la rama actual a la otra denominada *nombre*.
5. Primero es necesario posicionarse en la rama donde se busca que queden guardados los cambios. Luego, se escribe el comando “git merge *rama_nueva*”. Así, todo lo generado en la rama nueva se volcará en la rama donde nos posicionamos y escribimos el código mencionado.
6. Se crea utilizando el comando “-m '*mensaje*'” siendo *mensaje* lo que queremos aclarar en el commit para registrar los cambios.
7. Luego de haber guardado y descripto los cambios, para enviar un commit a GitHub se usa el comando “git push origin *main*”. Siendo *main* la rama principal aunque también se pueden enviar ramas individuales, no necesariamente el trabajo entero.
8. Es el espacio digital, cargado en los servidores de GitHub, donde se pueden subir los trabajos hechos en Git y compartirlos con otros programadores.
9. Es necesario utilizar el comando “git remote add *repositorio_remoto*”. Así, al repositorio creado en GitHub se le enviará el repositorio local creado, en este caso, con el nombre de “repositorio_remoto”.
10. A través del comando “git push origin *nombre_rama*”.
11. Usando el comando “git pull origin *nombre_rama*”
12. Un *fork* es una copia de un repositorio creada en una cuenta diferente a la propia permitiendo desarrollar cambios sin afectar al original. El *fork* se realiza creando una copia en la cuenta del usuario.
13. Para enviar una solicitud de extracción se usa el comando “git pull *nombre_repo*”. Así se descargarán los cambios y se fusionarán con el repositorio local.
14. Hay que revisar los cambios en GitHub, hacer click en “merge” y elegir el método de fusión.

15. Una etiqueta es una referencia fija que uno marca en el historial de commits para registrar algún cambio importante o destacable. Se aplican usando el comando “git tag”. Éstas se suben a GitHub usando el comando “git push --tags”.
16. El historial se puede revisar con el comando “git log --oneline”. Luego, para borrar el historial se utiliza el comando “git branch -D *nombre_rama*”
17. Un repositorio privado es aquel en el que únicamente el usuario creador tiene acceso. Éste puede compartir el link con otros usuarios sin la necesidad de que sea abierto a todo el público.
18. Cuando se crea el repositorio se puede seleccionar la opción para que sea privado.
19. El repositorio público es aquel al que cualquier persona con un usuario en GitHub puede acceder. Basta con compartir el link con las demás personas.

ACTIVIDAD 2.



Link del repo: <https://github.com/totomxs/Repo-TP2.git>

ACTIVIDAD 3.

Link del repositorio <https://github.com/totomxs/TP2-Actividad3.git>