

# PROGRAMACION I

## TP N°2

Dominguez Ivan

Comisión 13

- 1) Contestar las siguientes preguntas utilizando las guías y documentación proporcionada (Desarrollar las respuestas):

- **¿Qué es GitHub?**

GitHub es una plataforma de desarrollo colaborativo que potencia el uso de Git, dando la posibilidad de almacenar archivos y sus respectivos avances, en un servidor en la nube. Permite llevar la información almacenada en un repositorio local a uno remoto.

- **¿Cómo crear un repositorio en GitHub?**

Una vez que hayamos creado una cuenta en GitHub e ingresemos a la pagina principal, debemos dirigirnos a nuestro perfil, para esto, vamos a encontrar nuestra foto (si es que agregamos una) arriba a la derecha, haciendo clic en la imagen vamos a poder acceder a nuestro perfil.

Ya en nuestro perfil vamos a visualizar nuestros repositorios, si es que ya creamos alguno. En la barra superior vamos a tener la opción **“Create new”** representado con un símbolo +. Al hacer clic vamos a tener la opción **“New repository”**, le damos a esa opción y accedemos a la configuración de un nuevo repositorio. Debemos nombrarlo, no puede tener el mismo nombre que un repositorio que ya existe, podemos agregar opcionalmente una descripción que mencione para que se utilizara por ejemplo y si queremos que sea publico o privado.

Una vez tengamos las configuraciones de nuestra preferencia le damos al botón **“Create repository”** y listo.

- **¿Cómo crear una rama en Git?**

Dentro de la consola debemos ejecutar el comando **git branch** y el nombre de la rama que queremos crear. Por ejemplo, el comando podría quedar de la siguiente manera **git branch nueva\_rama**.

- **¿Cómo cambiar a una rama en Git?**

Si estamos en la rama main o master y quieres cambiar a otra rama debemos ejecutar **git checkout nueva\_rama**, si tomamos como ejemplo la rama que creamos en la pregunta anterior, pero luego de checkout debemos tipear el nombre de la rama a la que queremos cambiar

- **¿Cómo fusionar ramas en Git?**

Para fusionar ramas primero debemos estar ubicados en la rama a la que queremos traer el resto, sino debemos utilizar el comando checkout. Una vez ubicados en la rama que queremos debemos ejecutar **git merge {el nombre de la rama que queremos traer a la que estamos ubicados}**

- **¿Cómo crear un commit en Git?**

Para crear un commit en Git, primero se añaden los cambios al área de preparación con **git add .**, luego se ejecuta **git commit -m “Mensaje descriptivo del commit”** para registrar los cambios en el historial.

- **¿Cómo enviar un commit a GitHub?**

Para enviar un commit a GitHub, se usa `git push origin {nombre de la rama}`, asegurándose de estar autenticado y en la rama correcta

- **¿Qué es un repositorio remoto?**

Un repositorio remoto es una versión alojada en un servidor de un repositorio Git local, permitiendo colaboración entre múltiples desarrolladores.

- **¿Cómo agregar un repositorio remoto a Git?**

Para agregar un repositorio remoto, se usa `git remote add origin {URL_del_repositorio}` y luego `git push -u origin {nombre de la rama}` para establecer la conexión.

- **¿Cómo empujar cambios a un repositorio remoto?**

Se utiliza `git push origin {rama}` para enviar los cambios locales a un repositorio remoto.

- **¿Cómo tirar de cambios de un repositorio remoto?**

Se usa `git pull origin {rama}` para obtener y fusionar los cambios más recientes de la rama especificada.

- **¿Qué es un fork de repositorio?**

Un fork es una copia de un repositorio en otra cuenta de GitHub, permitiendo modificaciones sin afectar el original.

- **¿Cómo crear un fork de un repositorio?**

Para hacer un fork, se debe ir al repositorio en GitHub y hacer clic en el botón "Fork".

- **¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?**

Se debe hacer un fork, realizar los cambios y luego abrir una pull request desde la pestaña "Pull requests" en GitHub.

- **¿Cómo aceptar una solicitud de extracción?**

El propietario o colaborador del repositorio revisa la pull request y la fusiona si los cambios son aprobados

- **¿Qué es una etiqueta en Git?**

Una etiqueta es un marcador en un commit específico que permite identificar versiones importantes.

- **¿Cómo crear una etiqueta en Git?**

Se usa `git tag -a {nombre_de_la_etiqueta} -m "Descripción"` para crear una etiqueta.

- **¿Cómo enviar una etiqueta a GitHub?**

Se debe ejecutar `git push origin {nombre_de_la_etiqueta}` para enviar la etiqueta al repositorio remoto.

- **¿Qué es un historial de Git?**

El historial de Git muestra todos los commits realizados en un repositorio.

- **¿Cómo ver el historial de Git?**

Se usa `git log` para visualizar los commits con su información detallada.

- **¿Cómo buscar en el historial de Git?**

Se pueden buscar commits por mensaje con `git log --grep="palabra"` o por cambios con `git log -S "código"`.

- **¿Cómo borrar el historial de Git?**

El historial no se puede eliminar completamente sin perder los commits, pero se puede usar `git reset --hard {commit}` para volver a un estado anterior.

- **¿Qué es un repositorio privado en GitHub?**

Un repositorio privado es un repositorio al que solo pueden ser acceder usuarios autorizados.

- **¿Cómo crear un repositorio privado en GitHub?**

Primeramente hay que seguir los pasos de la segunda pregunta, al crear un nuevo repositorio en GitHub, se selecciona la opción `"Private"`

- **¿Cómo invitar a alguien a un repositorio privado en GitHub?**

Desde la pestaña `"Settings"` del repositorio, en `"Manage access"`, se pueden agregar colaboradores.

- **¿Qué es un repositorio público en GitHub?**

Un repositorio público es un repositorio al que puede acceder cualquier usuario de GitHub.

- **¿Cómo crear un repositorio público en GitHub?**

Se crea un repositorio en GitHub y se selecciona la opción `"Public"`.

- **¿Cómo compartir un repositorio público en GitHub?**

Se puede compartir la URL del repositorio para que otros puedan verlo o clonarlo.