

TRABAJO PRÁCTICO N°2

PROGRAMACIÓN I

Profesor/es: Julieta Trapé

Alumno/a: Karen Lauk

Actividades

1) Contestar las preguntas:

a. ¿Qué es GitHub?

Es una plataforma donde se alojan repositorios.

b. ¿Cómo crear un repositorio en GitHub? → Iniciar sesión en la página de GitHub.

- Hacer clic en “New” o “Create repository”
- Asignar un nombre
- Agregar una descripción (opcional) → Seleccionar si será público o privado.
- Puede inicializar con un archivo README
- Hacer clic en Create repository

c. ¿Cómo crear una rama en Git?

Para crear una rama utilizamos “git branch”

d. ¿Cómo cambiar a una rama en Git? Para cambiar a una rama utilizamos “git checkout”.

e. ¿Cómo fusionar ramas en Git? Para combinar ramas utilizamos “Git merge”.

f. ¿Cómo crear un commit en Git?

Para crear un commit utilizamos “git add.”

g. ¿Cómo enviar un commit a GitHub?

Para enviar un commit utilizamos “git push”

h. ¿Qué es un repositorio remoto?

Es un repositorio alojado en un servidor (GitHub) que permite compartir el código con otros colaboradores.

i. ¿Cómo agregar un repositorio remoto a Git?

Para agregar un repositorio remoto utilizamos “git remote add origin”.

j. ¿Cómo empujar cambios a un repositorio remoto? Para empujar cambios a un repositorio “git push origin”.

k. ¿Cómo tirar de cambios de un repositorio remoto?

Para tirar cambios a un repositorio utilizamos “git pull origin”.

l. ¿Qué es un fork de repositorio?

Un fork es una copia de un repositorio en tu cuenta de GitHub para hacer modificaciones sin afectar el original.

m. ¿Cómo crear un fork de un repositorio?

- Ir al repositorio
- Hacer clic en el botón **fork** (parte superior derecha).
- Se realizó la copia

n. ¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?

- Subir los cambios a GitHub en una nueva rama.
- En GitHub, vamos a la pestaña **pull requests**. → Hacer clic en **New pull requests**.
- Selecciona las ramas y hacer clic en **create pull request**.

o. ¿Cómo aceptar una solicitud de extracción?

Para aceptar una solicitud de extracción hacemos los siguiente pasos:

- Vamos a la pestaña **pull requests**. → Abrimos la solicitud y revisamos los cambios
- Hacemos clic en **merge pull requests**.
- Confirmamos la fusión de los cambios.

p. ¿Qué es una etiqueta en Git?

Es una referencia a un punto específico en la historia de un repositorio, generalmente usada para marcar versiones.

q. ¿Cómo crear una etiqueta en Git?

Para crear una etiqueta utilizamos "git **tag -a v1.0 -m**"

r. ¿Cómo enviar una etiqueta a GitHub?

Para enviar una etiqueta utilizamos "git **push origin v1.0**"

s. ¿Qué es un historial de Git?

Es un registro de todos los cambios y commits realizados en un repositorio.

t. ¿Cómo ver el historial de Git?

Para ver el historial utilizamos "git **log**"

u. ¿Cómo buscar en el historial de Git?

Para buscar el historial utilizamos "git **log -grep="palabra clave"**".

v. ¿Cómo borrar el historial de Git?

Para borrar el historial utilizamos el comando "git **rebase**"

w. ¿Qué es un repositorio privado en GitHub?

Es un repositorio al que solo las personas invitadas pueden acceder.

x. ¿Cómo crear un repositorio privado en GitHub?

En la consigna 2/b explica los pasos de creación, seleccionando qué será privado.

y. ¿Cómo invitar a alguien a un repositorio privado en GitHub?

Para invitar a alguien seguimos los siguientes pasos:

- Ve a la pestaña **settings** del repositorio.
- Selecciona **manage access**
- Agregar al usuario con su nombre de GitHub

z. ¿Qué es un repositorio público en GitHub? Es un repositorio accesible por cualquier usuario. **aa. ¿Cómo crear un repositorio público en GitHub?**

Lo mismo que en la consigna 2/b, a diferencia que el ejercicio **w**, ponemos público en vez de privado. **bb. ¿Cómo compartir un repositorio público en GitHub?**

Podemos compartir la URL del repositorio con otros.

2) Realizar la actividad

Crear repositorio - Agregar un archivo - Crea un branches

<https://github.com/karenlauk/prueba>

3) Realizar la actividad

- Crear un repositorio en github
- Clonar el repositorio a tu máquina
- Crea una rama y editar un archivo
- Volver a la rama principal y editar el mismo archivo
- Hacer un merge y generar un conflicto

- Resolver el conflicto
- Subir los cambios a GitHub
- Verificar en GitHub

<https://github.com/karenlauk/UTN-TUPaD-P1>