

Práctico 2: Git y GitHub

Actividades

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

- **¿Qué es GitHub?** GitHub es una plataforma web que permite almacenar y gestionar proyectos que utilizan Git como sistema de control de versiones. Facilita la colaboración entre desarrolladores y el seguimiento de cambios en el código.
- **¿Cómo crear un repositorio en GitHub?** Para crear un repositorio en GitHub, debemos iniciar sesión en nuestra cuenta, hacer clic en "New repository", asignarle un nombre, elegir si será público o privado, y finalmente hacer clic en "Create repository".
- **¿Cómo crear una rama en Git?** Para crear una rama en Git, se utiliza el comando: `git branch nombre-de-la-rama`
- **¿Cómo cambiar a una rama en Git?** Para cambiar de rama en Git, se usa el comando: `git checkout nombre-de-la-rama`
- **¿Cómo fusionar ramas en Git?** Para fusionar ramas, primero nos posicionamos en la rama principal y luego ejecutamos: `git merge nombre-de-la-rama`
- **¿Cómo crear un commit en Git?** Primero, agregamos los archivos con: `git add .` Luego, creamos el commit con: `git commit -m "Mensaje del commit"`
- **¿Cómo enviar un commit a GitHub?** Se utiliza el comando: `git push origin nombre-de-la-rama`
- **¿Qué es un repositorio remoto?** Es un repositorio alojado en un servidor externo, como GitHub, que permite compartir y colaborar en un proyecto.
- **¿Cómo agregar un repositorio remoto a Git?** Utilizamos el comando: `git remote add origin URL-del-repositorio`
- **¿Cómo empujar cambios a un repositorio remoto?** Se hace con el comando: `git push origin nombre-de-la-rama`
- **¿Cómo tirar de cambios de un repositorio remoto?** Para obtener cambios del repositorio remoto, usamos: `git pull origin nombre-de-la-rama`
- **¿Qué es un fork de repositorio?** Un fork es una copia de un repositorio que se guarda en nuestra cuenta de GitHub. Permite realizar cambios sin afectar el proyecto original.
- **¿Cómo crear un fork de un repositorio?** En la página del repositorio que queremos copiar,

hacemos clic en el botón "Fork" y seleccionamos nuestra cuenta.

- **¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?** Desde nuestro fork, hacemos clic en "New pull request", elegimos las ramas y enviamos la solicitud.
- **¿Cómo aceptar una solicitud de extracción?** El propietario del repositorio debe revisar la solicitud y, si está de acuerdo, hacer clic en "Merge pull request".
- **¿Qué es una etiqueta en Git?** Una etiqueta es un marcador que señala un punto específico en la historia del repositorio, normalmente para identificar versiones.
- **¿Cómo crear una etiqueta en Git?** Se utiliza el comando: `git tag nombre-de-la-etiqueta`
- **¿Cómo enviar una etiqueta a GitHub?** Con el comando: `git push origin nombre-de-la-etiqueta`
- **¿Qué es un historial de Git?** Es el registro de todos los commits realizados en un repositorio, mostrando los cambios, autores y fechas.
- **¿Cómo ver el historial de Git?** Se utiliza el comando: `git log`
- **¿Cómo buscar en el historial de Git?** Podemos buscar usando: `git log --grep="palabra-clave"`
- **¿Cómo borrar el historial de Git?** Se puede reiniciar el repositorio eliminando la carpeta .git y volviendo a inicializarlo con `git init`.
- **¿Qué es un repositorio privado en GitHub?** Es un repositorio que sólo puede ser visto y gestionado por el propietario y las personas que éste invite.
- **¿Cómo crear un repositorio privado en GitHub?** Al crear un repositorio, seleccionamos la opción "Private".
- **¿Cómo invitar a alguien a un repositorio privado en GitHub?** En la sección "Settings" del repositorio, vamos a "Collaborators" y agregamos el nombre de usuario de la persona.
- **¿Qué es un repositorio público en GitHub?** Es un repositorio visible para cualquier usuario de GitHub.
- **¿Cómo crear un repositorio público en GitHub?** Al crear un repositorio, seleccionamos la opción "Public".
- **¿Cómo compartir un repositorio público en GitHub?** Simplemente copiamos y compartimos la URL del repositorio.

2. Enlace de repositorio de la actividad realizada:

<https://github.com/AriGrela/TP2-Ejercicio2-UTN>

3. Enlace de repositorio de la actividad realizada:

<https://github.com/AriGrela/conflict-exercise>