

Actividades:

Actividad 1:

- 1) Github es una plataforma que almacena repositorios.
- 2) si tenemos una cuenta de github, hay que ir a la pantalla principal y tocar el botón “new”, después rellenar el campo del nombre de tu repositorio, con una descripción si se quiere, y presionar el botón verde de abajo.
- 3) con el comando: `git branch [nombre de rama]`
- 4) con el comando: `git checkout [nombre de rama]`
- 5) con el comando: `git merge [nombre de rama]`
- 6) con el comando: `git commit -m “[descripción de lo que hiciste]”`
- 7) con el comando: `git push origin [rama]`
- 8) un repositorio remoto es una copia de tu repositorio local que se encuentra en internet
- 9) con: `git remote add origin “[link del repositorio]”`
- 10) con el comando: `git push`
- 11) con: `git pull`
- 12) un fork es una copia de un repositorio ajeno en tu propia cuenta, independiente del original.
- 13) para hacer un fork tienes que buscar un repositorio ajeno, presionar el botón de arriba que dice “fork” y rellenar los datos para después presionar “create fork”
- 14) la pull request se habilita después de hacer un commit dentro de un repositorio forkeado, cuando entras a tu fork te aparecerá la opción para hacerla, pones un comentario y das al botón ver de abajo.
- 15) en el repositorio donde se encuentra la pull request. Presiona “pull request” en la parte superior del repo, anda a la pr que quieras aceptar y revisa los cambios, después presionar “review change”, seguido de “approve” y “submit review”, finalmente presiona “merge pull request” y confirma el merge.
- 16) las etiquetas son referencias que apuntan a puntos concretos en el historial de git
- 17) con el comando: `git tag [nombre]`
- 18) con el comando: `git push --tags`
- 19) el historial es un registro de los cambios realizados dentro de un proyecto
- 20) el historial se puede ver con el comando: `git log`
- 21) para buscar en el historial hay varias formas, para buscar palabras clave es: `git log --grep="[palabra]”`. Con `git log`, aparece una lista de commits con su autor, fecha y mensaje, por

los cuales también se puede buscar. Si es para buscar los cambios de un archivo específico es:
git log -- [nombre del archivo]

22) para borrar el historial hay muchas maneras, si es para borrar el historial local entero basta con borrar la carpeta oculta “.git”, otra es crear otra rama sin commits (git checkout --orphan), añadir todos los archivos ahí (add .), haces un commit, borras la rama anterior y renombras la nueva y por último git push -f

23) un repositorio privado es aquel en el que solo usuarios que estén autorizados por uno mismo pueden ver y hacer commits en el repositorio, será invisible para el resto.

24) un repositorio privado se crea como cualquier otro, con la diferencia de que cuando lo estas por crear tienes que seleccionar la opción “private” en vez de “public” y listo.

25) para invitar a alguien a un repo privado hay que estar dentro del repositorio e ir a la opción de configuración, después a la opción “colaboradores” y poner tu contraseña, buscar el nombre de la persona que quieres autorizar, lo seleccionas y presionas “agregar [usuario]”. Después el tiene que aceptar la invitación que le habrá llegado por correo.

26) un repositorio público es aquel que todos los usuarios pueden ver, y tu eliges quien puede hacer commits.

27) los repositorios se crean por defecto como público, así que no hay que hacer mas aparte de poner el nombre y crearlo, pero abajo están las opciones para cambiar entre público y privado.

28) para compartir un repositorio público puedes invitar a colaboradores como en el punto 25 o compartir la url manualmente.

Actividad 2:

Link del repositorio: https://github.com/Lucanms/repo_practico2

Actividad 3:

Link de repositorio: <https://github.com/Lucanms/conflict-exercise>