Práctico 2: Git y GitHub

1. GitHub es una plataforma para guardar y controlar versiones de repositorios, facilitando a desarrolladores a compartir y colaborar en proyectos controlando cambios en el mismo.
2. desde gitHub crear un repositorio con nombre y una descripcion, desde la terminal de gitbash y en la carpeta del proyecto, iniciar el repositorio con el comando git init, agregar los archivos con git add ., confirmar con git commit -m “” y un comentario de lo que se esta subiendo, y terminar con los comandos git remote add y git push para terminar de subir el repositorio.
3. para crear una rama en Git, usar el comando git branch y el nombre de la rama a crear.
4. para cambiar de rama se una el comando git checkout y el nombre de la rama a cambiar
5. para fusionar ramas se una git merge
6. para crear un commit, despues de agregar los cambios con git add, crear el commit con git commit -m “” y la descripcion de los cambios
7. para enviar el commit se una el comando git remote add y git push origin y el nombre de la rama
8. un repositorio remoto es un repositorio en un servidor para que diferentes desarolladores colaboren en el mismo proyecto viendo los cambios en una version del codigo comun
9. para agregar un repositorio remoto a git se una el comando git remote add origin <https://githib.com/usuario/nombre> del repositorio.git
10. con git push subimos los cambios al repositorio remoto, git push origin nombre
11. git pull origin nombre se usa para actualizar los cambios en un repositorio remoto
12. fork de repositorio es copiar un repositorio de otra persona en tu cuenta de github, se usa para trabajar en proyectos de codigo abierto sin afectar el original
13. para crear un fork hay que apretar en la opcion fork en el repositorio que quieras copiar y tener en tu cuenta, con git clone se puede clonar el fork en una pc propia
14. pull request se usa para sugerir cambios en un repositorio, despues de subir los cambios a github, en el repositorio hay una opcion para comparar y hacer pull request
15. Para aceptar un pull request, en la opcion de pull requests se pueden aceptar con la opcion merge pull request
16. una etiqueta en git se usa para identificar versiones del proyecto
17. una etiqueta en git se crea con el comando git tag -a version -m “”descripcion
18. Para enviar se usa el comando git push origin version
19. el historial de git sirve para tener un registro de los commits en un repositorio
20. con el comando git log se puede ver el historial de commits
21. para buscar en el historial un commit especifico se puede usar git log –greep=””
22. para borrar el historial de git se puede usar git-filter-repo
23. un repositorio privado solo lo pueden ver el propietario y las personas habilitadas, se usa en proyectos privados
24. al crear un nuevo repositorio, despues de crear una descripcion se puede elegir entre publico o privado
25. para invitar colaboradores, en el repositorio privado, en settings, esta la opcion de agregar personas, con nombre de usuario y correo se puede enviar invitacion
26. un repositorio publico es visible para cualquiera en github, pero solo ver, no se pueden hacer cambios
27. al crear un repositorio nuevo elegir la opcion publico en vez de privado
28. para compartir un repositorio publico hay que compartir la URL del repo