1.Crear un usuario:

git config --global user.name "name"

git config --global user.email "email"

------------------------------------------------------------------------------------------------------------------------

2.Ver las propiedades de git:

git config --list

------------------------------------------------------------------------------------------------------------------------

3.Pedir ayuda:

git help config

------------------------------------------------------------------------------------------------------------------------

4.Crear un repositorio:

git init

------------------------------------------------------------------------------------------------------------------------

5.Pasar al area de trabajo los archivos y para reastrear los arcivos:

git add . Para pasar todos los archivos que hay en el repositorio

git add "nombre" para pasar un archivo especifico

------------------------------------------------------------------------------------------------------------------------

6.Abrir el visual code:

code .

------------------------------------------------------------------------------------------------------------------------

7.Para ver los archivos en el repositorio y en el area de trabajo:

git status

git commit

------------------------------------------------------------------------------------------------------------------------

8. para clonar un repositorio.

git clone <nombre del repositorio>

------------------------------------------------------------------------------------------------------------------------

9.Para viajare ene el tiempo al commit que deseemos (a archivos, fotos, etc.):

git checkout -- <archivo>…

------------------------------------------------------------------------------------------------------------------------

9.1 para regresar al commit actual es:

git commit master

------------------------------------------------------------------------------------------------------------------------

10.para ignorar algun archivo:

.gitignore

------------------------------------------------------------------------------------------------------------------------

11.Para abrir el commit:

git commit

11.1.Para agregar un texto de que es lo que modificaste

git commit -m "texto"

------------------------------------------------------------------------------------------------------------------------

12. para comficar las modificaciones

git commit -a -m 'added new benchmarks'

------------------------------------------------------------------------------------------------------------------------

13.para sacar el archivo del area de preparacion:

rm <nombre>

y para eliminarlo completamente es:

git rm <nombre>

------------------------------------------------------------------------------------------------------------------------

14.para que git no lo rastre y no eliminarlo de disco dur es:

git rm --cached README

------------------------------------------------------------------------------------------------------------------------

15.Este comando elimina todos los archivos que acaben con ~.

git rm \\*~

------------------------------------------------------------------------------------------------------------------------

16.Cambiar nombre

git mv mentiris.txt readme.txt

------------------------------------------------------------------------------------------------------------------------

17.mirar los commits:

git log --oneline --all --graph --decorate

git log

------------------------------------------------------------------------------------------------------------------------

18.para deshacer una accmon es

git commit –amend

------------------------------------------------------------------------------------------------------------------------

19. git reset (mata a los commits)

19.1. git reset --soft <nombre del commit> (no toca el working area)

19.2. git reset --mixed (no toca el staging are ni el workign area)

19.3. git reset --hard (elimina todo)\*super peligroso\*

Ramificaciones:

head: es el commit en que estamos

rama: es una linea de tiempo en nuestro proyecto

rama master: es la rama por default

tipos de fusiones :

Fast-Forward: solo va a hacer la fusión, esto pasa cuando se trabaja con archivos diferentes o lineas de código distintas.

Manual-Merge: Antes de hacer la fusion tiene que pasar por niosotros, esto ocurre cuando se trabaja en los mismos archivos

20.para ver en que rama estoy

git branch

21.Crear una nueva rama:

git branch <nombre de la rama>

22. Pasara otra rama:

git checkout <nombre de la rama>

23. para eliminar una rama:

git branch -D <nombre de la rama>

24.para que una rama sea absorbida por la en que estemos:

git merge <nombre de la rama que vamos a fusionar>

25. crear y moverse a esa rama a la misma vez

git branch -b <nombre de la rama>

Github

26.es para conectar el proyecto local con el proyecto remoto:

git remote add origin <nombre del repositorio>

27.para ver los repositorios que están conectados

git remote -v

28. para eliminar los repositorios remotos es:

git remote remove origin

29,para enviar los archivos del local al remoto:

git push origin master