Para hacer seguimiento a una carpeta, debemos,

1. Clic derecho en git bash
2. Con

git init

ya comenzamos a hacer seguimiento

1. Con

git status -s

vemos el estado de los archivos en esa ubicación, incluyendo los que están en carpetas

1. Si queremos hacer seguimiento a un archivo en especifico usamos

git add intex.html

A Ejemplo1Archivo.html

?? css/

?? js/

El interrogante significa que no le hace seguimiento, la A ya le esta haciendo seguimiento, esta en el área de ensayo.

1. Ahora cuando queremos hacer una instantánea ya guardada usamos commit

git commit -m "Comienzo del proyecto"

1. Si intento aquí ver el status el index ha desaparecido del seguimiento
2. Si hacemos un cambio en el archivo llevado al repositorio y vemos status vemos que el archivo aparece con una m, diciendo que fue modificado, pero aún no ha sido respaldado.

$ git status -s

M Ejemplo1Archivo.html

?? Tutorial.docx

?? css/

?? js/

?? ~$torial.docx

1. Debo primero siempre llevar los archivos a un área de ensayo, cuando uso de nuevo Add, aparece el archivo modificado pero en verde, luego de eso si podemos guardar otra copia del archivo

$ git commit -m "Titulo y parrafo uno"

1. Si queremos ver las copias guardadas, lo hacemos con

$ git log --oneline

3f09ab6 (HEAD -> master) Titulo y parrafo uno

87c5d25 Comienzo del proyecto

1. Con la instrucción git reset --hard hacemos una restauración del archivo con el código de la instantánea

$ git reset --hard 87c5d25

HEAD is now at 87c5d25 Comienzo del proyecto

1. Para hacer seguimiento a todos los archivos

git add .

1. Ahora si quiero hacer un add y un commit al tiempo usamos

git commit -am “Archivos Css y Js”

1. Para modificar la descripción de algún commit vamos a modificar el editor vim

git commit –amend

luego :i y ahí comenzamos a editar y otras vainas raras

<https://www.youtube.com/watch?v=0UlqvTJzOL4&list=PLU8oAlHdN5BlyaPFiNQcV0xDqy0eR35aU&index=3>

1. Cuando creamos un repositorio en github buscamos el código

git remote add origin <https://github.com/AnalysiCo/CursoGit.git>

lo que hará que se supa todo ahí, creo que la primera vez toca escribir las 3 líneas.

$ git branch -M main

$ git push -u origin main

1. Traer modificaciones, ahora traermos la instrucción pull

git pull

1. Las tags permiten agregar versiones terminadas del proyecto

$ git tag 12-11-21version1 -m "Version 1 del Proyecto"

$ git push –tags

1. Para clonar damos git clone y la url que corresponda
2. Para crear una nueva rama o Branch

<https://www.youtube.com/watch?v=q9LJIHDgJtE&list=PLU8oAlHdN5BlyaPFiNQcV0xDqy0eR35aU&index=5>

$ git branch javascript

1. Para ver en que rama estamos

$ git Branch

1. Para movernos de rama

git checkout "javascript"

1. Para fusionar dos ramas hay que movernos a la rama master, hay que hacer merge

git merge javascript

1. Si queremos dejar de usar una rama, deberíamos borrarla, con

Git Branch -d Javascript