# TUTORIAL PARA TONTITOS COMO YO

1. Descargar Git : <https://git-scm.com/>
2. Abrimos la consola (Windows +cmd) y escribimos lo sgtes comandos reemplazando nuestros datos:

(los errores más comunes son, por ejemplo, escribir un solo menos antes de global, no respetar los espacios en blanco exactos, no dar enter al completar un comando. )

\*git config --global –user.name “PEPITO MEIBEN ROMERO COMPILADOR”

\*git config --global --user.email “[alomacho@gmail.com](mailto:alomacho@gmail.com)” (no olvidarse las comillas)

-- config: indica a git que vamos a modificar un elemento de configuración.

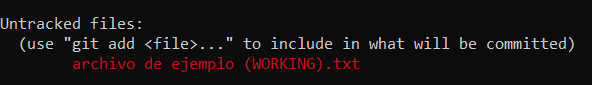
--global: el elemento de configuración se va a cambiar de forma global(en toda la compu) y se usará cómo valor por defecto cuando no exista de forma local.

3)si ya temenos esto configurado lo podemos ver con un el sgte comando:

git config –global --list

4)(opcional) se recomienda ejecutar el siguiente comando por una cuestión visual, los archivos que tengamos que agregar al staging area estarán en rojo y los archivos que ya agregamos al staging área listos para commitear estén en verde.

Ejemplo:



Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

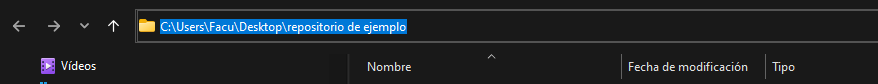
El comando es : git config --global color.ui a

u

Podemos cerrar el cmd tranquilamente.to

5) creamos una carpeta que utilizaremos para crear nuestro repositorio local, da igual la ubicación o el nombre.

6)seleccionamos la dirección de la carpeta y en su lugar escribimos cmd y damos enter, ejemplo:



Rectángulo

Descripción generada automáticamente con confianza media + enter

Tendría que aparecer algo como esto:

Texto

Descripción generada automáticamente

Aún no hay ningún repositorio, para esto debemos escribir en el cmd que acabamos de abrir el comando:

git init

seguido de enter. Debería aparecer algo así :

Texto

Descripción generada automáticamente

Ahí nos indica que inicializamos un repositorio de git que por el momento está vacío, en esa dirección.

Para confirmar que creamos un repositorio de git, en la carpeta donde lo inicializamos hay un archivo oculto que se llama .git

Para poder verlo vamos a vista y tildamos la opción ver archivos ocultos en Windows 10 o en Windows 11 ver y mostrar elementos ocultos y deberían poder visualizar una carpetita así:



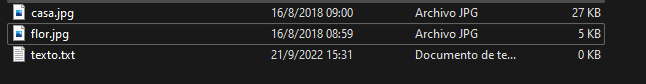
Esa carpeta no debe ser tocada.

creamos o ponemos archivos dentro del repositorio que queremos crear, y con el comando :

git status

podremos visualizarlo en el área de working ósea en rojo.

Ejemplo:



Ejecutamos el git status: (con este comando podemos ver el estado en el que se encuentran nuestros archivos)

Texto

Descripción generada automáticamente

Y efectivamente aparecen, también nos dice que usemos git add para agregarlos al staging área, entonces lo vamos a hacer.

Ejecutamos:

git add .

(no olvidarse el punto, el punto agrega todos los archivos a lo macho)

Hacemos un git status para ver que pasó.

Texto

Descripción generada automáticamente

Nos dice que hay cambios para comitear

Ejecutamos el commit:

git commit . -m “mensaje descriptivo de lo que realizamos”

Texto

Descripción generada automáticamente

Ahora si hacemos un git status (sin modificar nada ni cambiar nada) este debería estar vacío

Texto

Descripción generada automáticamente

El comando git commit confirma los cambios del área del stage y los guarda en el repositorio, el -m se usa para escribir un mensaje que describa lo que estemos haciendo en ese commit**.**

**El historial de commits se pueden ver con el comando:**

**git log**

**veamos:**

**Texto

Descripción generada automáticamente**

**Podríamos decir que con el comando git log podemos ver un historial de las modificaciones, quien las hizo, donde , fecha y hora exacta.**