

## TABLA DE COMANDOS EN GITBASH

Comando	Descripción
<b>Crear</b>	
\$ mkdir	Crear carpeta o directorio.
\$ touch	Crear archivo.
\$ git init	Crear o Inicializar un repositorio local.
\$ git clone "URL de repositorio remoto"	Para descargar por primera vez un repositorio remoto a nuestro computador.
<b>Mostrar</b>	
\$ ls	Listar archivos en el directorio o carpeta.
\$ ls -r	Mirar todo el contenido de las carpetas con sus respectivos archivos.
\$ pwd	Identificar la carpeta en la que se está trabajando actualmente.
\$ ls -a	Muestra los archivos y/o carpetas ocultas
<b>Eliminar</b>	
\$ rmdir	Eliminar un carpeta o directorio (Git solo permite eliminar la carpeta si esta vacia)
\$ rm	Eliminar archivos
\$ rm -r	Eliminar carpeta con contenido adentro.
<b>Navegar</b>	
\$ cd	Cambiar de carpeta o directorio (con este comando nos podemos mover entre distintos directorios).
\$ cd .. /	Ir hacia atrás en la terminal
<b>Cambios locales</b>	
\$ git status	Para ver el estado de nuestros archivos
\$ git add .	Para indicar que queremos agregar todos los archivos al repositorio.
\$ git add "nombre del archivo"	Para indicar que queremos agregar un lo archivo al repositorio.
\$ git commit -m "mensaje"	Para confirmar que queremos subir de manera definitiva aquellos archivos que agregamos
<b>Historial de commits</b>	
\$ git log	Mostrar todos los commits, comenzando por el más nuevo. Registra el historial de nuestros cambios en el proyecto
<b>Agregar identidad al repositorio</b>	
\$ git config user.name	Agregar nombre de mi usuario de Github
\$ git config user.email	Agregar e-mail registrado en Github

<b>\$ git config –global user.name</b>	Todos los repositorios existentes en mi computador, ya tendrán mi usuario y e-mail.
<b>\$ git config –global user.email</b>	
<b>\$ git remote -v</b>	Verificar que el repositorio local quedó conectado con el repositorio remoto
<b>\$ git config –list</b>	Para conocer qué repositorio remoto se está usando. Al final de contenido se puede observar el usuario y email de github.
<b>Actualización y publicación</b>	
<b>\$ git push origin main</b>	Para enviar los archivos del repositorio local al repositorio remoto
<b>\$ git pull origin main</b>	Para poder actualizar los archivos de nuestro repositorio local con respecto a los que están en el repositorio remoto
<b>Ramas</b>	
<b>\$ git branch “nombre de la rama”</b>	Crear una nueva rama
<b>\$ git branch -d “nombre de la rama”</b>	Eliminar una rama. Git evita que eliminemos una rama si tiene cambios que aun no se han fusionado con la rama Main.
<b>\$ git branch -D “nombre de la rama”</b>	Fuerza la eliminación de la rama especificada, incluso si tiene cambios.
<b>\$ git checkout “nombre de la rama”</b>	Se usa para moverse de una rama a otra. Git solo permitirá que nos movamos a otra rama si NO tenemos cambios. Si tenemos primero debemos eliminarlos ó confirmarlos.
<b>\$ git push origin “nombre de la rama”</b>	Subir cambios a una rama.
<b>\$ git pull origin “nombre de la rama”</b>	Traer o descargar cambios de una rama.
<b>Varios</b>	
<b>\$ git --version</b>	Conocer qué versión de Git tenemos instalada y verificar que la instalación de Git se haya realizado correctamente.
<b>\$ git merge “nombre de la rama”</b>	Integrar <rama> en el HEAD actual.
<b>\$ clear</b>	Limpia la pantalla de la terminal.
<b>\$ nano</b>	Escribir texto dentro de un archivo o editar el contenido del archivo.
<b>\$ cat</b>	Leer la información que el archivo tiene adentro.