

Comandos	Tarea clase 6 Marco Ossandon
Crear repositorio local	<p><b>Git init //</b> crea un repositorio local.</p> <p><b>Git config user.name +"Nombre" //</b> agrega nombre de usuario.*</p> <p><b>Git config user.email +"email" //</b> agrega correo electrónico de usuario.*</p> <p>*Debe ser el user y mail de cuenta Github.</p> <p><b>git config --global user.name "nombreDeUsuario" //</b></p> <p><b>git config --global user.email "email" //</b></p> <p>Para configurar nuestra identidad e email de manera global y no tener que aclarar continuamente nuestra identidad.</p> <p><b>**Para comprobar que los los datos que ingresamos, podemos utilizar los mismos comandos pero hasta <b>user.name</b> (si queremos comprobar el usuario) y hasta <b>user.email</b> (para comprobar email).</b></p>
Conectar/vincular repositorio local con remoto (Github)	<p><b>git remote add origin +URL de repositorio remoto //</b> para vincular repositorio local con el remoto.</p> <p><b>Git remote -v //</b> sirve para comprobar si efectivamente se vinculó.</p> <p><b>Git clone "url de github" //</b> nos permite crear una copia del repositorio remoto en nuestro computador.</p>
Subir archivos	<p><b>Git add +"nombre de archivo" //</b> agrega archivo específico a staging area previo al repositorio.</p> <p><b>Git add . //</b> agrega todos los archivos untracked al staging area del repositorio.</p> <p><b>Git commit -m "breve explicación del cambio realizado" //</b> lo utilizamos para confirmar que queremos subir los archivos previamente guardados (<b>added</b>) al repositorio. En las comillas va el nombre del "commit", que responde a breve descripción de las modificaciones que hicimos.</p> <p><b>Git push origin main //</b> nos permite subir definitivamente el "commit" al repositorio remoto.</p> <p><b>Git push origin +Nombre de branch //</b> lo usamos para subir archivos editados dentro de alguna de las ramas creadas del repositorio.</p>
Descarga de archivos	<b>Git pull //</b> descargar los commits guardados en el repositorio remoto.
Ramificaciones	<p><b>Git branch "nombre nuevo" //</b> nos permite crear una nueva ramificación del repositorio.</p> <p><b>Git checkout "nombre de rama" //</b> nos permite posicionarnos en la rama nueva para efectuar acciones desde allí.</p> <p><b>Git merge //</b> funciona para hacer un "merge" entre desde una rama o repositorio hacia la ubicación en la que nos encontramos.</p>
Navegación	<p><b>Git status //</b> nos muestra el estado del repositorio, esencialmente para saber si quedan archivos "untracked" o para confirmar que todos los archivos se encuentran "added" al staging area, lo que nos da pie a continuar con el proceso de "commit".</p> <p><b>Git log //</b> muestra el historial del repositorio.</p> <p><b>git log -- "ruta del archivo" //</b> historial de archivo específico.</p> <p><b>git log --author=usuario //</b> historial de usuario particular.</p>