**COMANDOS DE GIT**

|  |  |
| --- | --- |
| **Comando** | **Descripción** |
| **git help *comando*** | Permite observar lo que realiza el comando |
| **git init** | Crea un repositorio en el directorio actual |
| **git clone** url | Clona un repositorio remoto dentro de un directorio |
| **git add** path | Adiciona un archivo o un directorio de manera recursiva |
| **git rm** ruta | Remueve un archivo o directorio del árbol de trabajo        -f : Fuerza la eliminación de un archivo del repositorio |
| **git mv** origen destino | Mueve el archivo o directorio a una nueva ruta        -f : Sobre-escribe los archivos existentes en la ruta destino |
| **git checkout** [rev] archivo | Recupera un archivo desde la rama o revisión actual        -f : Sobre-escribe los cambios locales no guardados |
| **git status** | Imprime un reporte del estado actual del árbol de trabajo local |
| **git diff** [ruta] | Muestra la diferencia entre los cambios en el árbol de trabajo local |
| **git diff** HEAD ruta | Muestra las diferencias entre los cambios registrados y los no registrados |
| **git add** path | Selecciona el archivo para que sea incluido en el próximo commit |
| **git reset** HEAD ruta | Marca el archivo para que no sea incluido en el próximo commit |
| **git commit** | Realiza el commit de los archivos que han sido registrados (con git-add)        -a : Automáticamente registra todos los archivos modificados |
| **git reset** --soft HEAD^ | Deshace commit & conserva los cambios en el árbol de trabajo local |
| **git reset** --hard HEAD^ | Restablece el árbol de trabajo local a la versión del ultimo commit |
| **git clean** | Elimina archivos desconocidos del árbol de trabajo local |
| **git log** [ruta] | Muestra el log del commit, opcionalmente de la ruta especifica |
| **git log** [desde [..hasta]] | Muestra el log del commit para un rango de revisiones dado        --stat : Lista el reporte de diferencias de cada revisión        -S'pattern' : Busca el historial de cambios que concuerden con el patrón de búsqueda |
| **git blame** [archivo] | Muestra el archivo relacionado con las modificaciones realizadas |
| **git fetch** [remote] | Trae los cambios desde un repositorio remoto |
| **git pull** [remote] | Descarga y guarda los cambios realizados desde un repositorio remoto |
| **git push** [remote] | Guarda los cambios en un repositorio remoto |
| **git remote** | Lista los repositorios remotos |
| **git remote add** remote url | Añade un repositorio remoto a la lista de repositorios registrados |
| **git checkout** rama | Cambia el árbol de trabajo local a la rama indicada        -b rama : Crea la rama antes de cambiar el árbol de trabajo local a dicha rama |
| **git branch** | Lista las ramas locales |
| **git branch** -f rama rev | Sobre-escribe la rama existente y comienza desde la revisión |
| **git merge** rama | Guarda los cambios desde la rama |
| **git apply** - < archivo | Aplica el parche desde consola (stdin) |
| **git format-patch** desde [..hasta] | Formatea un parche con un mensaje de log y un reporte de diferencias (diffstat) |
| **git archive** rev > archivo | Exporta resumen de la revisión (snapshot) a un archivo        --prefix=dir/ : Anida todos los archivos del snapshot en el directorio        --format=[tar|zip] : Especifica el formato de archivo a utilizar: tar or zip |
| **git tag** name [revision] | Crea una etiqueta para la revisión referida        -s : Firma la etiqueta con su llave privada usando GPG        -l [patrón] : Imprime etiquetas y opcionalmente los registros que concuerden con el patrón de búsqueda |
| **git bisect** | encontrar por la búsqueda binaria el cambio que introdujo un error |
| **git grep** | líneas de impresión que coincidan con un patrón |
| **git rebase** | puerto frontal local realiza la actualización de la cabeza de subida |
| **git show** | muestra varios tipos de objetos |