Comandos para iniciar un repositorio y enlazar con un servidor externo

Comando	Descripción
`git init`	Inicializa un repositorio.
`git remote add origin urlServer`	Enlaza el proyecto con la siguiente ruta para sincronizar cambios.
`git push -u origin master`	Sube los cambios a la urlServer indicada.
Estos comandos solamente se ejecutan una sola vez por proyecto para su inicialización	

`git clone urlRepositorio`
 	
`La url va sin doble comillas.`
	Clona un repositorio del servidor.

Comandos comunes (Sincronización de cambios)

`git status`	Lista los archivos del proyecto y sus estados posibles. • Pendiente de agregar. • Agregado. • Modificado • Eliminado.
`git add NombreArchivo`	Agrega un archivo específico para realizar seguimiento.
`git add .`	Agrega todos los archivos para realizar seguimiento.

`git commit -m "Mensaje"`
 	
O
 	
`Git commit -am "Mensaje"`
Agrega todos los archivos que se encuentran con modificaciones, por lo que es posible saltarse el comando "add .".
 	
En el caso que el archivo no se encuentre con seguimiento, se deberá ejecutar "add ." previamente.
	Registra un estado del proyecto asociado a lo registrado en "git add"

`git pull origin`
 	
o
 	
`git pull`
(Toma como implícito la referencia a origin)
	Baja los cambios en el servidor.
	
	Se recomienda siempre hacer un pull antes de un push.
	
	La acción de realizar un pull seguido de un push tiene el nombre de "Sync".

`git push origin`
 	
o
 	
`git push`
(Toma como implícito la referencia a origin)
	Sube los cambios a un servidor remoto. Los cambios subidos son los registrados en los commits.

Comandos de restitución de información

Comando	Descripción
`git reset --hard`	Pierde todos los cambios no comprometidos. Esto quiere decir que restaura el proyecto a la versión del último commit registrado.

`git reset --hard idCommit`
 	
En caso accidental de ejecutar el comando anterior, use los siguientes comandos para restaurar:
`git reflog show`
`git reset HEAD@{1}`
 	
`Si desea continuar con la restauración utilice los siguientes comandos para actualizar el historial (Esto eliminará los commits del servidor).`
`git clean -f`
`git push -f`
	Regresa a un commit indicando su Id. Elimina todos los cambios realizados hasta llegar a ese commit.
	
	Recomendación: Utilizar comando **`git log --oneline`** para obtener el listado de Ids.
	
	Este comando aparte de deshacer también puede rehacer. Solamente es necesario indicar el Id del commit al que se quiere restaurar.
	
	Este comando se considera peligroso.

`git reset --soft HEAD~1.` Todo el proyecto se mantiene igual, el commit es eliminado.	En el caso que cree un commit accidental de forma local, puede eliminarlo sin alterar el estado del proyecto.
`git checkout -- NombreArchivo.Extensión`	Deshace todos los cambios realizado a un archivo.

Comandos de información y estados

Comando	Descripción
`git log --oneline`	Muestra en una línea los commit realizados.
`git log --oneline --decorate --all --graph`	Muestra en una línea los commit realizado, con una ayuda visual.
`git reflog`	Muestra el historial de commit, incluyendo el historial de commit eliminados. Esto es solamente información local del equipo.
`git status`	Lista los archivos del proyecto y sus estados posibles. • Pendiente de agregar. • Agregado. • Modificado • Eliminado.

Comandos de Tags

`git tag NombreVersion -m "Descripción"`
 	
Después de crear recuerde usar el siguiente comando para subir al servidor.
`git push --tag`
	Crea un tag asociado al último commit.

`git tag -d NombreTag`	Borra un Tag local.
`git push --delete origin NombreTag` **Observación:** Este comando es el mismo para eliminar una rama, por lo que borrará el elemento con el mismo valor de nombre. Se sugiere tener precaución.	Elimina un Tag del servidor.
`git tag -a NombreTag IdCommit -m "Descripción"`	Crea un tag asociado a un Id.
`git tag`	Lista los tags.
`git show NombreTag`	Muestra información sobre el Tag.

Comandos de gestión de ramas

Comando	Descripción
`git branch`	Lista todas las ramas asociadas a un proyecto.
`git checkout NombreRama`	Cambia la rama actual por la ingresa en NombreRama.
`git checkout -b NombreRama`	Cambia la rama actual por la ingresa en NombreRama. En el caso que no exista es creada.

`git branch -d NombreRama`
 	
`-d: En el caso que la rama no se encuentre fusionada el proceso de detendrá.`
 	
`-D: Borra la rama esté o no esté fusionada a la rama master.`
	Elimina una rama local.
	Para realizar esta eliminación es necesario estar en otra rama.

`git push --delete origin NombreRama`	Elimina una rama remota. Para realizar esta eliminación es necesario estar en otra rama.
`git merge NombreRamaACopiar`	En la rama que uno se encuentra traerá los cambios de NombreRamaACopiar.

Configuración de datos Globales

Comando	Descripción
`git config --global user.name "Nombre"`	Registra Nombre de Desarrollador. Puede ser cualquier nombre.
`git config --global user.email "email@ejemplo.com"`	Registra el nombre del correo que aparecerá asociado a los commits.

Configuración de salto de línea (LF/CRLF)

`git config core.autocrlf` **`true`**	Si trabajará solamente en Windows.
`git config core.autocrlf` **`false`**	Si trabajará en diferentes sistemas operativos. Mantenga la opción en true para que Git se encargue.
https://help.github.com/en/github/using-git/configuring-git-to-handle-line-endings	

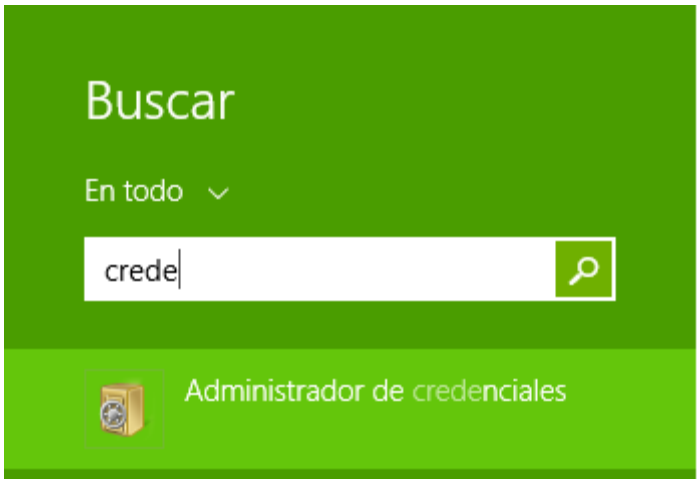
Borrar registro de cuenta en computador

Ejecute la siguiente lista de comandos:

git config --global --unset user.name	Borra el nombre registrado.
git config --global --unset user.email	Borra el correo registrado.
git config --global --unset credential.helper	Borra las credenciales almacenadas.

Si después de haber ejecutado dichos 3 comandos aún puede realizar push, intente borrando las credenciales de Windows.

1. Escriba en el buscador, credenciales y presione enter.



2. Seleccione “Credenciales de Windows”.
3. Busque en el listado las credenciales de git.
4. Seleccione la opción de “Quitar”.

En el caso que se olvide de realizar estos pasos, cambie su contraseña desde la página web.

Recuerde borrar sus repositorios locales en caso de estar en un equipo que no es suyo.