

## Comandos para iniciar un repositorio y enlazar con un servidor externo

Comando	Descripción
<code>git init</code>	Inicializa un repositorio.
<code>git remote add origin urlServer</code>	Enlaza el proyecto con la siguiente ruta para sincronizar cambios.
<code>git push -u origin master</code>	Sube los cambios a la urlServer indicada.
<b>Estos comandos solamente se ejecutan una sola vez por proyecto para su inicialización</b>	
<code>git clone urlRepositorio</code>	Clona un repositorio del servidor.

## Comandos comunes (Sincronización de cambios)

<code>git status</code>	Lista los archivos del proyecto y sus estados posibles. <ul style="list-style-type: none"><li>• Pendiente de agregar.</li><li>• Agregado.</li><li>• Modificado</li><li>• Eliminado.</li></ul>
<code>git add NombreArchivo</code>	Agrega un archivo específico para realizar seguimiento.
<code>git add .</code>	Agrega todos los archivos para realizar seguimiento.
<code>git commit -m "Mensaje"</code>  0  <code>Git commit -am "Mensaje"</code> (Ya no es necesario ejecutar el comando "git add" antes)	Registra un estado del proyecto asociado a lo registrado en "git add"
<code>git pull origin</code>	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".
<code>git push origin</code>	Sube los cambios a un servidor remoto. Los cambios subidos son los registrados en los commits.

## Comandos de restitución de información

Comando	Descripción
<code>git reset --hard idCommit</code>  Complementar con Reinicio de historial. <code>git clean -f</code> <code>git push -f</code>	Regresa a un commit anterior indicando su Id. Elimina todos los cambios realizados hasta llegar a ese commit.  <b>Recomendación:</b> Utilizar comando <b>git log --oneline</b> 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 peligrosa.
<code>git checkout --</code> NombreArchivo.Extensión	Deshace todos los cambios realizado a un archivo.

## Comandos de información y estados

Comando	Descripción
<code>git log --oneline</code>	Muestra en una línea los commit realizados.
<code>git log --oneline --decorate --all --graph</code>	Muestra en una línea los commit realizado, con una ayuda visual.
<code>git status</code>	Lista los archivos del proyecto y sus estados posibles. <ul style="list-style-type: none"> <li>• Pendiente de agregar.</li> <li>• Agregado.</li> <li>• Modificado</li> <li>• Eliminado.</li> </ul>

## Comandos de gestión de ramas

Comando	Descripción
<code>git branch</code>	Lista todas las ramas asociadas a un proyecto.
<code>git checkout NombreRama</code>	Cambia la rama actual por la ingresa en NombreRama.
<code>git checkout -b NombreRama</code>	Cambia la rama actual por la ingresa en NombreRama. En el caso que no exista es creada.
<code>git branch -d NombreRama</code>	Elimina una rama local. Para realizar esta eliminación es necesario estar en otra rama.
<code>git push --delete origin NombreRama</code>	Elimina una rama remota. Para realizar esta eliminación es necesario estar en otra rama.
<code>git merge NombreRamaACopiar</code>	En la rama que uno se encuentra traerá los cambios de NombreRamaACopiar.

## Configuración de datos Globales

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

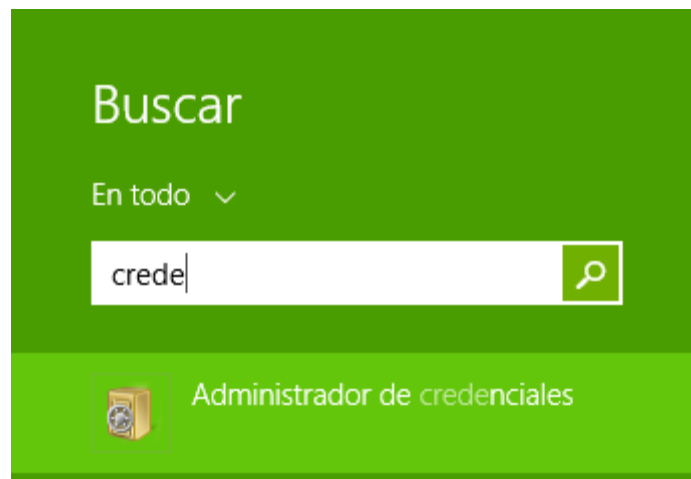
## Borrar registro de cuenta en computador

Ejecute la siguiente lista de comandos:

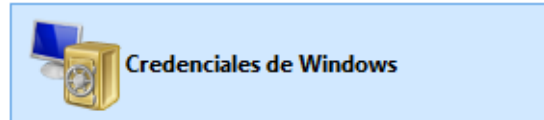
<code>git config --global --unset user.name</code>	Borra el nombre registrado.
<code>git config --global --unset user.email</code>	Borra el correo registrado.
<code>git config --global --unset credential.helper</code>	Borra las credenciales almacenadas.

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”.



[Copia de seguridad de credenciales](#) [Restaurar credenciales](#)

Credenciales de Windows

[Agregar una credencial de Windows](#)

No hay credenciales de Windows.

Credenciales basadas en certificados

[Agregar una credencial basada en certificado](#)

No hay certificados.

Credenciales genéricas

[Agregar una credencial genérica](#)

git:https://

Fecha de modificación:



Dirección de red o Internet: git:https://

Nombre de usuario:

Contraseña:

Persistencia: Equipo local

Editar

[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.**