|  |  |
| --- | --- |
| **Git init** | **Crear un nuevo repositorio de git en una carpeta.** |
| **Git clone** | **Crea una copia local del repositorio.** |

FLUJO DE TRABAJO

Tu repositorio local está compuesto por tres "árboles" administrados por git. El primero es tu **Directorio de trabajo** que contiene los archivos, el segundo es el **Index** que actúa como una zona intermedia, y el último es el **HEAD** **que apunta al último commit realizado.**

|  |  |
| --- | --- |
| **git add <filename>**  **git add .** | Puedes registrar cambios (añadirlos al Index). Este es el primer paso en el flujo de trabajo básico. |
| **git commit -m "Commit message"** | Ahora el archivo esta incluido en el HEAD de tu copia local, pero aún no en tu repositorio remoto. |
| **git push origin master** | Para enviar estos cambios a tu repositorio remoto. (Reemplaza master por la rama a la que quieres enviar tus cambios). |
| **git remote add origin <server>** | Si no has clonado un repositorio ya existente y quieres conectar tu repositorio local a un repositorio remoto. |
| **git checkout -b feature\_x** | Crea una nueva rama llamada "feature\_x" y cámbiate a ella. |
| **git checkout master** | Vuelve a la rama principal. |
| **git branch -d feature\_x** | Borra la rama. |
| **git push origin <branch>** | Una rama nueva no estará disponible para los demás a menos que subas (push) la rama a tu repositorio remoto. |

ACTUALIZA & FUSIONA

|  |  |
| --- | --- |
| **git pull** | Para actualizar tu repositorio local al commit más nuevo (para bajar y fusionar los cambios remotos). |
| **git merge <branch>** | Para fusionar otra rama a tu rama activa (por ejemplo master). En ambos casos git intentará fusionar automáticamente los cambios. |

En ambos casos git intentará fusionar automáticamente los cambios. Desafortunadamente, no siempre será posible y se podrán producir conflictos. Tú eres responsable de fusionar esos conflictos manualmente al editar los archivos mostrados por git.

|  |  |
| --- | --- |
| **git add <filename>** | Después de modificarlos, necesitas marcarlos como fusionados con… |
| **git diff <source\_branch> <target\_branch>** | Antes de fusionar los cambios, puedes revisarlos usando … |

ETIQUETAS

Se recomienda crear etiquetas para cada nueva versión publicada de un software. Este concepto no es nuevo, ya que estaba disponible en SVN. Puedes crear una nueva etiqueta llamada 1.0.0 ejecutando **git tag 1.0.0 1b2e1d63ff**

**1b2e1d63ff** se refiere a los 10 caracteres del commit id al cual quieres referirte con tu etiqueta.

Puedes obtener el commit id con **git log**,también puedes usar menos caracteres que el commit id, pero debe ser un valor único.

REEMPLAZA CAMBIOS LOCALES

En caso de que hagas algo mal, puedes reemplazar cambios locales usando el comando

**git checkout -- <filename>**

Este comando reemplaza los cambios en tu directorio de trabajo con el último contenido de HEAD. Los cambios que ya han sido agregados al Index, así como también los nuevos archivos, se mantendrán sin cambio.

Por otro lado, si quieres deshacer todos los cambios locales y commits, puedes traer la última versión del servidor y apuntar a tu copia local principal de esta forma

**git fetch origin**

**git reset --hard origin/master**

<http://rogerdudler.github.io/git-guide/index.es.html>

https://www.adictosaltrabajo.com/2011/04/26/github-first-steps-upload-project/