

git - la guía sencilla

una guía sencilla para comenzar con git. sin complicaciones ;)

Tweet

por Roger Dudler (traducido por @lfbarragan y @adrimatellanes)

créditos a @tfnico, @fhd y Namics

disponible en english, deutsch, français, indonesian, italiano, nederlands, polski, português, русский, türkçe,

မြန်မာ, 日本語, 中文, 한국어

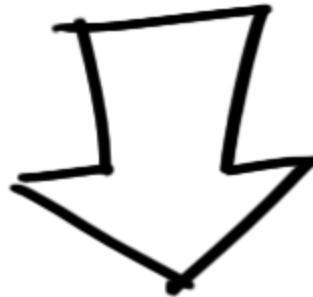
por favor, reporta cualquier problema en github



download the
cheat sheet
now. it's free!



want a simple
but powerful
git client for
your mac?



configuración

Descarga git para OSX

Descarga git para Windows

Descarga git para Linux

crea un repositorio nuevo

Crea un directorio nuevo, ábrelo y ejecuta

```
git init
```

para crear un nuevo repositorio de git.

hacer checkout a un repositorio

Crea una copia local del repositorio ejecutando

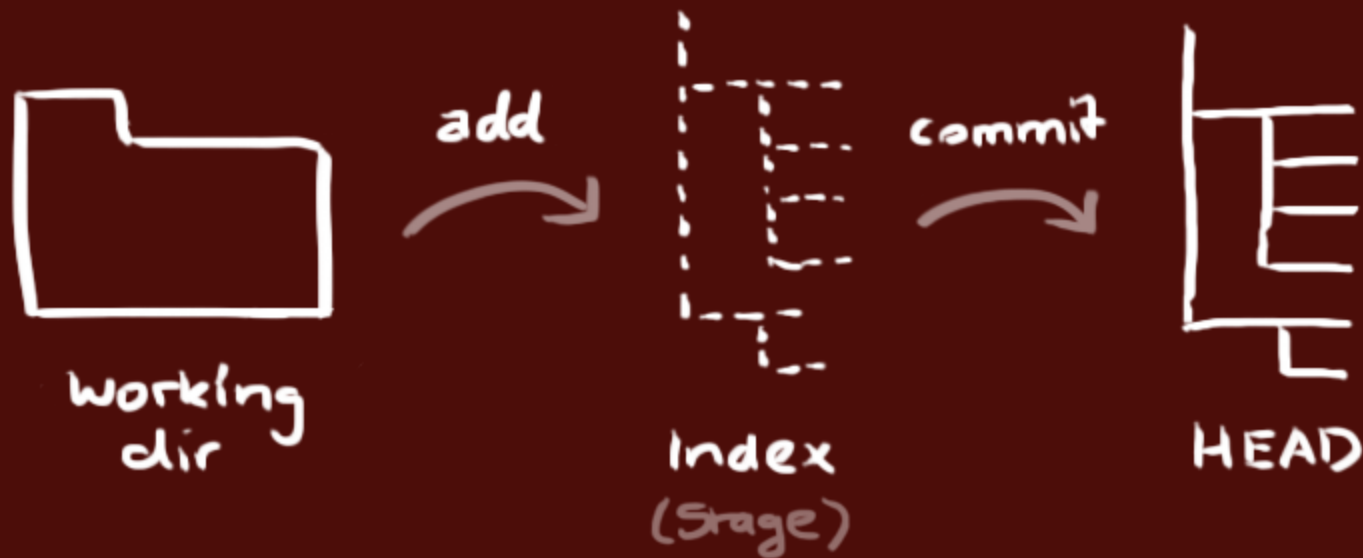
```
git clone /path/to/repository
```

Si utilizas un servidor remoto, ejecuta

```
git clone username@host:/path/to/repository
```

flujo de trabajo

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



add & commit

Puedes registrar cambios (añadirlos al **Index**) usando

```
git add <filename>
```

```
git add .
```

Este es el primer paso en el flujo de trabajo básico. Para hacer commit a estos cambios usa

```
git commit -m "Commit message"
```

Ahora el archivo esta incluido en el **HEAD**, pero aún no en tu repositorio remoto.

envío de cambios

Tus cambios están ahora en el **HEAD** de tu copia local. Para enviar estos cambios a tu repositorio remoto ejecuta

```
git push origin master
```

Reemplaza *master* por la rama a la que quieres enviar tus cambios.

Si no has clonado un repositorio ya existente y quieres conectar tu repositorio local a un repositorio remoto, usa

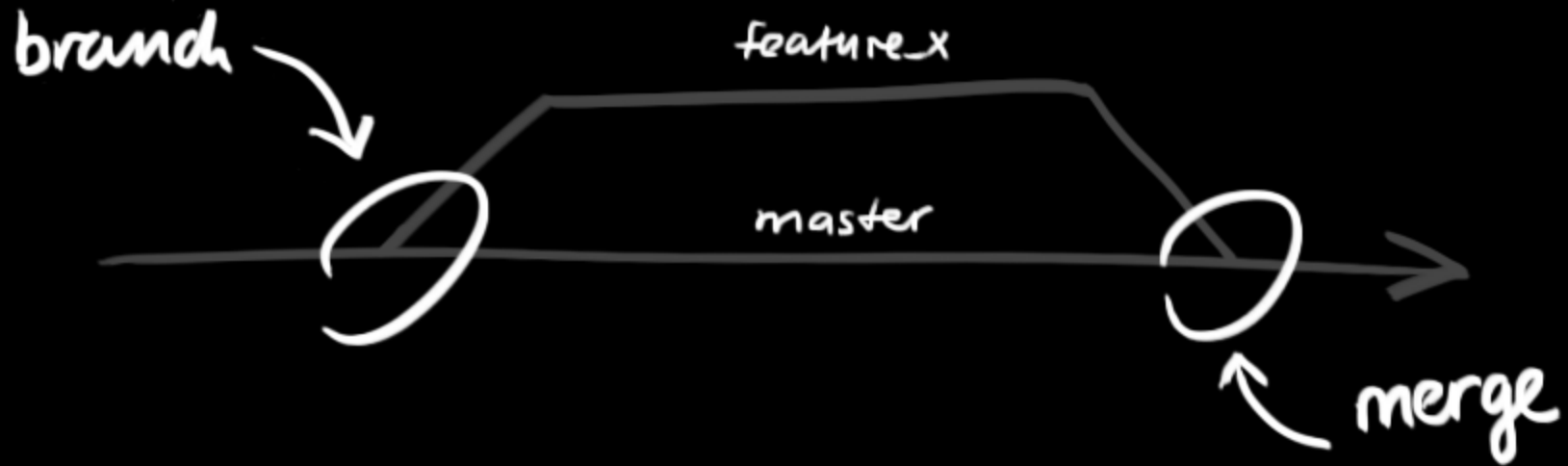
```
git remote add origin <server>
```

Ahora podrás subir tus cambios al repositorio remoto seleccionado.

ramas

Las ramas son utilizadas para desarrollar funcionalidades aisladas unas de otras. La rama *master* es la rama "por defecto" cuando creas un repositorio.

Crea nuevas ramas durante el desarrollo y fusi3nalas a la rama principal cuando termines.



Crea una nueva rama llamada "feature_x" y c3mbiate a ella usando

```
git checkout -b feature_x
```

vuelve a la rama principal

```
git checkout master
```


y borra la rama

```
git branch -d feature_x
```

Una rama nueva *no estará disponible para los demás* a menos que subas (push)

la rama a tu repositorio remoto

```
git push origin <branch>
```

actualiza & fusiona

Para actualizar tu repositorio local al commit más nuevo, ejecuta

```
git pull
```

en tu directorio de trabajo para *bajar* y *fusionar* los cambios remotos.

Para fusionar otra rama a tu rama activa (por ejemplo master), utiliza

```
git merge <branch>
```

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. Después de modificarlos, necesitas marcarlos como fusionados con

```
git add <filename>
```

Antes de fusionar los cambios, puedes revisarlos usando

```
git diff <source_branch> <target_branch>
```

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 (lo que seguramente nunca suceda ;) 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
```

datos útiles

Interfaz gráfica por defecto

```
gitk
```

Colores especiales para la consola

```
git config color.ui true
```

Mostrar sólo una línea por cada commit en la traza

```
git config format.pretty oneline
```

Agregar archivos de forma interactiva

```
git add -i
```

enlaces & recursos

clientes gráficos

GitX (L) (OSX, open source)

Tower (OSX)

Source Tree (OSX, free)

GitHub for Mac (OSX, free)

GitBox (OSX)

guías

Git Community Book

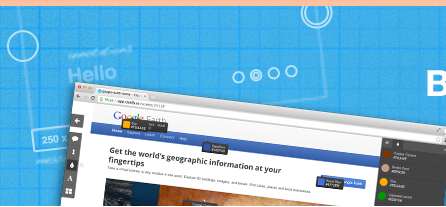
Pro Git

Think like a git

GitHub Help

A Visual Git Guide

Clarify



Building websites? Working with Designers? Try Clarify.

A Project by Roger Dudler, Author of The Git Simple Guide



comentarios

221 Comments

git - the simple guide

 Login ▾

 Recommend 95

 Tweet

 Share

Sort by Newest ▾



Join the discussion...



LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name



José González • a month ago

Buenísimoooo!!! Al fin un tutorial de GIT entendible.

^ | v • Reply • Share ›



Kike Man • a month ago

Realmente es el mejor tutorial que he encontrado!!

^ | v • Reply • Share ›



Ryuuzaki • 2 months ago

Buenas a todos, soy nuevo en esto de git, he instalado un servidor googs y tengo una imagen montada en este servidor, que de dicha imagen hice algo asi como una copia en otro direcrio, de forma que me deje revisar codigo de ese proyecto mientras yo sigo trabajando en la otra imagen en el mismo proyecto y no afectar el de la copia, aqui va mi duda, como yo puedo actualizar (Pull) mi copia una vez que le de subir (Push) en mi estacion de trabajo, es decir caundo yo modifique algo y le de push quisiera que automaticamente se reflejara en el directorio de la copia...gracias de antemano

^ | v • Reply • Share ›



Alex Bustos ➔ Ryuuzaki • 20 days ago

Dependiendo de tu servidor Git (GitHub/GitLab/Bitbucket, entre otros), lo mejor sería trabajar CI/CD, de tal manera que tu servidor Git ejecute test que validen tus commits y una vez validados, se encargue (de forma manual con unos pocos clicks, o automatizada) de subir los cambios al servidor de produccion, de desplegar.

Ahora, para responderte la pregunta, tendrías que en el servidor Git, usar su API o revisar triggers, ante cada commit... O hacer un script que automaticamente haga pull cada cierto periodo de tiempo, pero formas hay varias y quizas mejores que las que he mencionado

^ | v • Reply • Share ›



Kike Man ➔ Ryuuzaki • a month ago

Uelall! comparando un poco tu situacion... y desde mi punto de vista no creo que

Hola!! comprendo un poco tu situación, y desde mi punto de vista no creo que sea posible, debido a que tu copia pertenece a otro directorio y eso GIT lo tomaria como si fuese otro Repositorio. Mira si puedes anexar las rutas de como tienes tus servidor googs y de como tienes tu copia te puedo ayudar e encontrar alguna forma de solucionar tu problema

^ | v • Reply • Share >



John Fabian Torres • 2 months ago

I love your post bro

^ | v • Reply • Share >



Nestor Penuela • 2 months ago

Buenos dias, lo que sucede es que tengo git sobre linux sobre el puesto 10080, y no me funciona la url de acceso, el servidor esta arriba, pero ha tenido problemas, me puede alguien recordar como inicio el servicio de git en el servidor?
Gracias!!!

^ | v • Reply • Share >



Miguel ➔ Nestor Penuela • 2 months ago

con "git init" y para la url usa el https

^ | v • Reply • Share >



Begoña Herrero • 3 months ago

Hola ! tengo un problema, no sé como lo hice pero en la terminal me pone siempre, desde que la abro git (master) como lo quito? por lo que he leído, he hecho un repositorio globa

^ | v • Reply • Share >



Alex Bustos ➔ Begoña Herrero • 20 days ago

En algun lado hay una carpeta .git . Bastaría con matarla

^ | v • Reply • Share >



Alex Matías Bravo Inostroza ➔ Begoña Herrero • 2 months ago

usa git checkout para cambiar rama, si no es el caso borra la que no quieras usar.

^ | v • Reply • Share >



Jhon Gesell Villanueva • 4 months ago

Bien muy amable, gracias por esta introducción inicial al mundo Git.

^ | v • Reply • Share >



Ernesto Enrique Flames • 5 months ago

Que buena guía, gracias por tomarte el tiempo de hacerla. Saludos.

^ | v • Reply • Share ›



MireYaka • 5 months ago

Muy útil! Gracias!

^ | v • Reply • Share ›



Pablo Grr.. • 6 months ago

Hermoso

^ | v • Reply • Share ›



Sergio • 6 months ago

Muchas gracias!

^ | v • Reply • Share ›



Alejandro • 6 months ago

Muchas gracias.

^ | v • Reply • Share ›



Óscar A. Montiel • 7 months ago

Excelente amigo. Muchas gracias.

^ | v • Reply • Share ›



Afu Tse (Chainiz) • 7 months ago

Excelente!!

^ | v • Reply • Share ›



Jonatan Vera • 8 months ago

Excelente documento!!!!

^ | v • Reply • Share ›



Ricky • 8 months ago

Excelente trabajo. Gracias.

^ | v • Reply • Share ›



Juan Carlos Heredia Mayer • 8 months ago

Buen trabajo, gracias por compartirlo.

13 ^ | v • Reply • Share ›



Carlos Rivero • 8 months ago

Hermano mio. Muchas gracias. Excelente.

^ | v • Reply • Share ›



Wenubell • 8 months ago

la letra es enorme, pero mil gracias, me has salvado la vida bien salvada jaja gracias

^ | v • Reply • Share ›



Diego Santiago • 8 months ago

alguien tiene alguna idea de porque mis commits siguen siendo subidos con el usuario del dueño anterior de la computadora con la que ahora trabajo?. Ya cambie el user local y global y sigue asi

^ | v • Reply • Share ›



Javier Herrera ➔ Diego Santiago • 8 months ago

Hola Diego, porque la persona que usaba la computadora dejó su nombre de usuario global. Tenés que sobre escribir el mismo así:

```
git config --global user.name "John Doe"
```

```
git config --global user.email "johndoe@example.com"
```

Lógicamente reemplazando por tu usuario y correo que usás para GitHub o similar.

^ | v • Reply • Share ›



Diego Santiago ➔ Javier Herrera • 8 months ago

Que tal, si de hecho ya cambie el nombre de usuario local y global y el correo electronico pero en el repositorio continuan apareciendo los commits con el otro nombre-

^ | v • Reply • Share ›



Luis Orellana • 9 months ago

Hola a todos.

: Alguien sabe porque en la etapa de crear un branch, el tutorial dice que hay que eliminar

¿Alguien sabe porque en la etapa de crear un branch, el tutorial dice que hay que eliminar la rama creada?... me refiero cuando menciona:

y borra la rama

```
git branch -d feature_x
```

^ | v • Reply • Share ›



Luis Fernando García Pérez ➔ Luis Orellana • 9 months ago

eso es solo si ya no necesitas, la rama. no tienes por que borrarla si no quieres

1 ^ | v • Reply • Share ›



Dante Alvarez Tupiño • 10 months ago

Preciso ;)

^ | v • Reply • Share ›



MyNamelsDick ➔ Dante Alvarez Tupiño • 9 months ago

yess dick head

^ | v • Reply • Share ›



Ray Carreno • 10 months ago

Es una buena guia de inicio para proyectos pequeños y poco complejos.

^ | v • Reply • Share ›



alex sanchez • a year ago

Fácil, útil y bonito.

^ | v • Reply • Share ›



Andy Hernández ➔ alex sanchez • 10 months ago

Como madre le entiendes?

Yo no entiendo como hacer ni madre :/

Como monto la app de fondo negra a la maquina? Que se usa

Es que enserio nadie dice como chi... aserlo?

^ | v • Reply • Share ›



Vega-Yed ➔ Andy Hernández • 9 months ago

para que quieres instalar git si ni sabes usar una terminal

^ | v • Reply • Share ›



MyNamelsDick → Andy Hernández • 9 months ago

Tienes pinta de latino tercer mundista.

^ | v • Reply • Share ›



Inho Scar → MyNamelsDick • 3 months ago

que comentario mas xenofobico

^ | v • Reply • Share ›



Latina_tercermundista → MyNamelsDick • 7 months ago

Esta aprendiendo, nadie nace sabiendo todo

^ | v • Reply • Share ›



alex sánchez → Andy Hernández • 10 months ago

Hola compañero, veras el fondo negro del que hablas en realidad es solo la terminal de comandos que trae cualquier PC o Mac. En windows se llama CMD y en Mac se le conoce como terminal. Si tienes windows en el menu de inicio escribe solo "cmd" y te aparecera de inmediato. Suerte :)

1 ^ | v • Reply • Share ›



Reinaldo La Rosa • a year ago

Excelente guía rápida.

^ | v • Reply • Share ›



Cancrexo • a year ago

Gracias compañero!

Muy claro y conciso!

Salu2

^ | v • Reply • Share ›



DIEGO SORA • a year ago

Excelente!! Muchas Gracias.

^ | v • Reply • Share ›



carlos mar • a year ago

gracias

^ | v • Reply • Share ›



Miguel Ángel Aparici • a year ago

Hacía tiempo que no veía algo tan simple, conciso y a la vez utilísimo!!

Muchas gracias!!

1 ^ | v • Reply • Share >



Ioderik gutierrez • a year ago

Me hiciste el dia bro, gracias

^ | v • Reply • Share >



Jesus Rivera • a year ago

Excelente bro! Gracias por la guia!

^ | v • Reply • Share >



Naguara Avizor • a year ago

:o ... Ohh.. lo del brach y el merge lo entendi perfecto con esa imagen <3 thanks!

^ | v • Reply • Share >



Sebas • a year ago

Excelente, es la mejor guia de cualquier cosa que alla visto

^ | v • Reply • Share >



Fernando Moreno Ruiz • a year ago

superbien explicado <3

^ | v • Reply • Share >



Dev Rodriguez • a year ago

Sencillo mas no simple, que buena explicacion.

1 ^ | v • Reply • Share >