

git - guia prático

apenas um guia prático para começar com git. sem complicação ;)



por Roger Dudler

créditos para @tfnico, @fhd and Namics

english, deutsch, español, français, indonesian, italiano, nederlands, polski, русский,

□□□□□□, 日本語, 中文, 한국어

por favor informe problemas em [github](#)



download the
cheat sheet
now. it's free!



want a simple
but powerful
git client for
your mac?

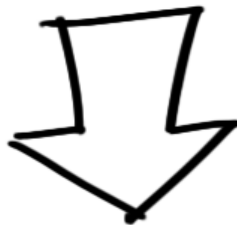


Are You a Front-End Developer?

by Roger Dudler, Author of the Git Simple Guide

Try Frontify

Now Free with
[Github Integration!](#)



instalação

Baixe o git para OSX

Baixe o git para Windows

Baixe o git para Linux

criando um novo repositório

crie uma nova pasta, abra-a e execute o comando

```
git init
```

para criar um novo repositório.

obtenha um repositório

crie uma cópia de trabalho em um repositório local executando o comando

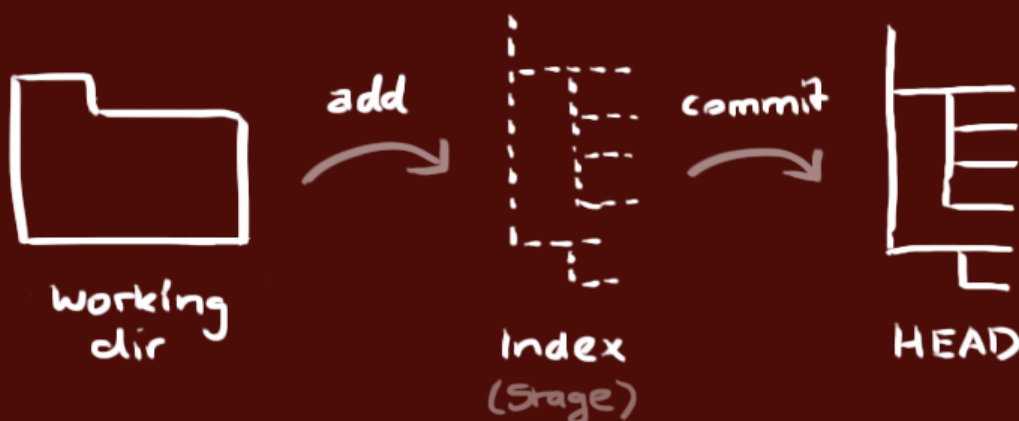
```
git clone /caminho/para/o/repositório
```

quando usar um servidor remoto, seu comando será

```
git clone usuário@servidor:/caminho/para/o/repositório
```

fluxo de trabalho

seus repositórios locais consistem em três "árvores" mantidas pelo git. a primeira delas é sua **Working Directory** que contém os arquivos vigentes. a segunda **Index** que funciona como uma área temporária e finalmente a **HEAD** que aponta para o último *commit* (confirmação) que você fez.



adicionar & confirmar

Você pode propor mudanças (adicioná-las ao **Index**) usando

```
git add <arquivo>
```

```
git add *
```

Este é o primeiro passo no fluxo de trabalho básico do git. Para realmente confirmar estas mudanças (isto é, fazer um *commit*), use

```
git commit -m "comentários das alterações"
```

Agora o arquivo é enviado para o **HEAD**, mas ainda não para o repositório remoto.

enviando alterações

Suas alterações agora estão no **HEAD** da sua cópia de trabalho local.

Para enviar estas alterações ao seu repositório remoto, execute

```
git push origin master
```

Altere *master* para qualquer ramo (*branch*) desejado, enviando suas alterações para ele.

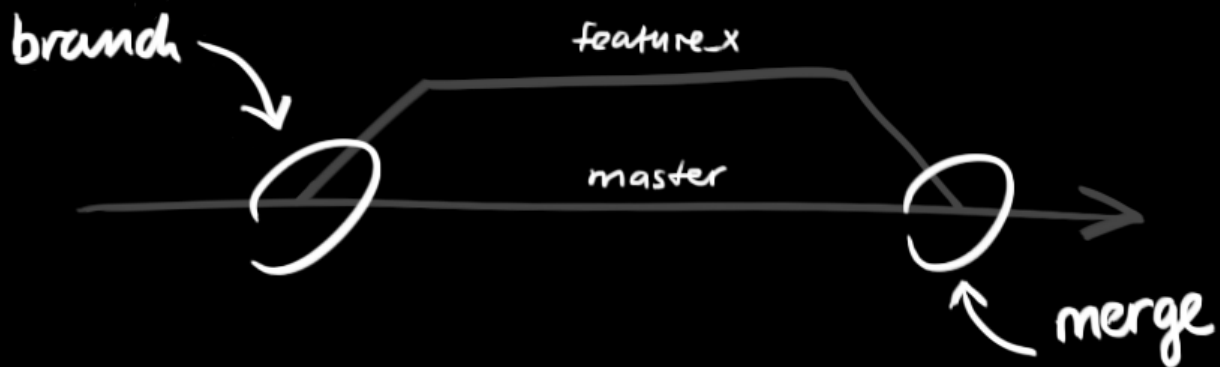
Se você não clonou um repositório existente e quer conectar seu repositório a um servidor remoto, você deve adicioná-lo com

```
git remote add origin <servidor>
```

Agora você é capaz de enviar suas alterações para o servidor remoto selecionado.

ramificando

Branches ("ramos") são utilizados para desenvolver funcionalidades isoladas umas das outras. O branch *master* é o branch "padrão" quando você cria um repositório. Use outros branches para desenvolver e mescle-os (*merge*) ao branch master após a conclusão.



crie um novo branch chamado "funcionalidade_x" e selecione-o usando

```
git checkout -b funcionalidade_x
```

retorne para o master usando

```
git checkout master
```

e remova o branch da seguinte forma

```
git branch -d funcionalidade_x
```

um branch *não está disponível a outros* a menos que você envie o

branch para seu repositório remoto

```
git push origin <funcionalidade_x>
```

atualizar & mesclar

para atualizar seu repositório local com a mais nova versão, execute

```
git pull
```

na sua pasta de trabalho para *obter e fazer merge* (mesclar) alterações remotas.

para fazer merge de um outro branch ao seu branch ativo (ex. master),

use

```
git merge <branch>
```

em ambos os casos o git tenta fazer o merge das alterações automaticamente. Infelizmente, isto nem sempre é possível e resulta em

conflitos. Você é responsável por fazer o merge estes *conflitos* manualmente editando os arquivos exibidos pelo git. Depois de alterar, você precisa marcá-los como merged com

```
git add <arquivo>
```

antes de fazer o merge das alterações, você pode também pré-visualizá-

as usando

```
git diff <branch origem> <branch destino>
```

rotulando

é recomendado criar rótulos para releases de software. Este é um conhecido conceito, que também existe no SVN. Você pode criar um novo rótulo chamado *1.0.0* executando o comando

```
git tag 1.0.0 1b2e1d63ff
```

o *1b2e1d63ff* representa os 10 primeiros caracteres do id de commit que você quer referenciar com seu rótulo. Você pode obter o id de commit com

```
git log
```

você pode também usar menos caracteres do id de commit, ele somente precisa ser único.

sobrescrever alterações locais

No caso de você ter feito algo errado (que seguramente nunca acontece ;) você pode sobrescrever as alterações locais usando o commando

```
git checkout -- <arquivo>
```

isto substitui as alterações na sua árvore de trabalho com o conteúdo mais recente no HEAD. Alterações já adicionadas ao index, bem como novos arquivos serão mantidos.

Se ao invés disso você deseja remover todas as alterações e commits locais, recupere o histórico mais recente do servidor e aponte para seu branch master local desta forma

```
git fetch origin
```

```
git reset --hard origin/master
```

dicas úteis

Interface gráfica padrão

```
gitk
```

usar saídas do git coloridas

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

exibir log em apenas uma linha por commit

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

fazer inclusões interativas

```
git add -i
```

recursos & links

clientes gráficos

GitX (L) (OSX, código aberto)

Tower (OSX)

Source Tree (OSX, gratuito)

GitHub for Mac (OSX, gratuito)

GitBox (OSX)

guias

Livro da comunidade Git

Pro Git

Pense como um git

Ajuda do GitHub

Um guia visual do Git

comentários

Oferta Especial LG OLED TV e G4

Oferta Especial. Compre uma LG OLED TV e ganhe um LG G4.
Promoção válida 08/06/16

[Learn More](#)

Sponsored by Pontofrio



162 Comments

git - the simple guide

[Login](#)

[Recommend](#) 73

[Share](#)

[Sort by Newest](#)



Join the discussion...



Junior Oliveira • 2 days ago

Muito Obrigado

[^](#) | [v](#) • [Reply](#) • [Share](#)



Pedro Henrique Ferreira Fonec • 4 days ago

Estava precisando de um tutorial simplificado como esse. Vlw d+!!

[^](#) | [v](#) • [Reply](#) • [Share](#)



Igor Guilherme • 12 days ago

Muito obrigado! Simples e objetivo, parabéns!

[^](#) | [v](#) • [Reply](#) • [Share](#)



italochesley • 2 months ago

sempre que eu preciso fazer algum commit recorro a esse tutorial...muito foda!
Parabéns!

[^](#) | [v](#) • [Reply](#) • [Share](#)



Erikson Magno • 2 months ago

Muito bom. Valeu mesmo

[^](#) | [v](#) • [Reply](#) • [Share](#)



John • 2 months ago

Obrigado! Bem detalhado.

[^](#) | [v](#) • [Reply](#) • [Share](#)



Nayara Valadares • 2 months ago

Muito bom!! obrigada :D

[^](#) | [v](#) • [Reply](#) • [Share](#)

**Caio Cesar** • 2 months ago

Muito Bom!
Obrigado!

^ | v • Reply • Share ›

**Jackson** • 3 months ago

Parabéns pelo tutorial muito bom!

Instalei o GitLab no servidor, sou iniciante, criei um repository, como faço para jogar meus commits em uma pasta /var/www/meuprojeto, esta pasta fica dentro do servidor apache, assim que eu fizer a alteração eu possa acessar os arquivos através da url dele www.meudominio.com/meuprojeto.

Alguem pode me dar um help?

^ | v • Reply • Share ›

**Yves Silva Romeiro** • 3 months ago

Voce é o cara me ajudo muito !!!!!

^ | v • Reply • Share ›

**Eris R. Jr** • 3 months ago

Muito bacana.

*O link para download/instruções de instalação no Linux agora é:
[https://git-scm.com/download/l...](https://git-scm.com/download/linux)

^ | v • Reply • Share ›

**Anderson** • 3 months ago

Melhor tutorial de git da internet, sem dúvida! Valeu!

^ | v • Reply • Share ›

**Te** • 3 months ago

simplesmente perfeito! não ajudou só eu, como todos do time de desenvolvimento da empresa favoritaram..rsrs

^ | v • Reply • Share ›

**Jones** • 3 months ago

Simplesmente demais! Valeu muito pela forma clara e método de fazer armazenar as informações de uma vez por todas..... OBRIGADU

^ | v • Reply • Share ›

**Marcelo Candido** • 3 months ago

Excelente!

^ | v • Reply • Share ›

**beatz** • 4 months ago

Perfeito, muito obrigado

^ | v • Reply • Share ›

**Jader Barros Pereira** • 4 months ago

Perfeito, lindo, inestimável e incomensurável o valor deste guia. Meus parabéns ao autor :)

^ | v • Reply • Share ›

**TiagoCeo** • 4 months ago

Esse guia salvou, eu tava usando o cliente TortoiseGit legal, mais que saber, vou usar é na linha de comando mesmo, simples e eficiente, valeu demais!!!!

^ | v • Reply • Share ›

**Willian Lhorente** ➔ **TiagoCeo** • 4 months ago

Particularmente, gosto bastante do TortoiseGit para Windows, no trabalho uso ele, pois agiliza o trabalho, principalmente quando quero adicionar/commitar somente alguns arquivos.

No PC de casa, uso por linha de comando para aprender trabalhar sem o

tortoise, caso precise.

^ | v • Reply • Share ›



Jhonata Matias • 5 months ago

Melhor guia dentre todos os guias que já vi. Sensacional!

^ | v • Reply • Share ›



David Oliveira • 5 months ago

que guia incrível! Com certeza essa foi uma das experiências mais inesquecíveis que eu tive com guias! Sensacional, brilhante.

^ | v • Reply • Share ›



Beto Silva • 5 months ago

Sensacional. Congrats

^ | v • Reply • Share ›



Valdecir Fernandes Marcia • 5 months ago

Simple, porém muito eficiente! Valeu!

^ | v • Reply • Share ›



Maurício Eugênio • 5 months ago

gostoso

^ | v • Reply • Share ›



John Gomes • 5 months ago

Muito bom!

^ | v • Reply • Share ›



brodrigues1990 • 5 months ago

Perfeito !

^ | v • Reply • Share ›



Felipe Genuino • 6 months ago

Lindo!

^ | v • Reply • Share ›



Adriano Francisco dos Santos • 7 months ago

Guia perfeito para iniciantes.
Ótimo trabalho.

1 ^ | v • Reply • Share ›



Roberto Vinicius • 7 months ago

Existe algum tuto com GIT GUI ??

^ | v • Reply • Share ›



Junior • 8 months ago

Parabéns, ótimo guia!

^ | v • Reply • Share ›



wdarking • 8 months ago

Me ajudou!

^ | v • Reply • Share ›



Jackson Jorge • 9 months ago

Finalmente um guia prático e decente para aprender a utilizar o git... Parabéns!

^ | v • Reply • Share ›



Jônatas de Paula • 9 months ago

















ctrl+D valeu!

2 ^ | v • Reply • Share ›



Igor De Souza Moraes • 9 months ago

melhor guia que achei até hoje

- 2 ^ | v • Reply • Share ›
-  **George Santos** • 10 months ago
Show de bola!
^ | v • Reply • Share ›
-  **Dorival Querino da Silva** • 10 months ago
Excelente guia!! Obrigado por compartilhar/traduzir/disponibilizar. Vlw!!
^ | v • Reply • Share ›
-  **Thiago Rizzo** • 10 months ago
Muito bom, parabéns e obrigado!
^ | v • Reply • Share ›
-  **Cassius Giorgio** • 10 months ago
Simples e direto ao ponto, extremamente útil! Valeu!!!
^ | v • Reply • Share ›
-  **marlonjke** • 10 months ago
Muito bom. Obrigado.
^ | v • Reply • Share ›
-  **Matioli Rei** • 10 months ago
SOBRE A SIMONE: EU CONCORDO COM O MATIOLI.
^ | v • Reply • Share ›
-  **Diego Ritzel** • a year ago
Obrigado! Site extremamente útil!
^ | v • Reply • Share ›
-  **matiolis2** • a year ago
@SIMONE, CONCORDO COM O MATIOLI
^ | v • Reply • Share ›
-  **Anman** • a year ago
O Matioli ta usando isso!
^ | v • Reply • Share ›
-  **manoe!** → Anman • a year ago
Verdade Cara!
^ | v • Reply • Share ›
-  **BRITÃO** → manoe! • a year ago
PODEROSO GIT
^ | v • Reply • Share ›
-  **Jman** • a year ago
That's Nice bro!
^ | v • Reply • Share ›
-  **Renato** • a year ago
Ótimo, parabéns!
^ | v • Reply • Share ›
-  **Flavio Roncat** • a year ago
Muito bom. Ajudou muito. Parabens!
^ | v • Reply • Share ›
-  **Fernando Araújo Bagno da Silva** • a year ago
Sem dúvidas o melhor tutorial a respeito que já li. Parabéns ao autor!
^ | v • Reply • Share ›
-  **alexsouza80** • a year ago

parabéns, sem enrolação e vai no que interessa!

^ | v • Reply • Share ›

Load more comments

 [Subscribe](#)

 [Add Disqus to your site](#)

[Add Disqus](#)

[Add](#)

 [Privacy](#)



Sponsored Links

36 Terrifying Military Photos
GeekVIP

7 Portuguese Words You'll Struggle To Pronounce (If You're Not Brazilian)
Babbel

The Planet's Richest Actors
The 405

The 35 Biggest Dogs In The World
PressRoomVIP

Buyer Beware: The 10 Lowest Rated Vehicles For 2015
Carophile

Revolutionary Way to Stop Snoring
My Snoring Solution