



# Git

📅 Created	@March 1, 2022 10:28 PM
👤 Created by	
🏷 Tags	Introdução
☰ Property	

Estudar: Chamada de pilha (funções dentro de funções)

[Livro Pro Git](#)

git flow → várias linhas do tempo.

<https://comandosgit.github.io/>

Git → Aula 1 de 4

Sistema de controle de versão → guarda dados (histórico)

Configurações: git —version

```
git config —global user.name ' '
```

```
git config — global user.email ' '
```

Git local e suas areas:

mkdir 'pasta' → diretório

cd → entrar no diretório

ls -a → listar

clear → limpa o código do git

echo 'texto1' >> arquivo\_1.txt

rm -rf nomearquivo → exclui arquivo ou pasta

git status → informação sobre o git

git init → iniciar um diretório .git onde ficará a versão de código

Areas:

Untracked : Não monitorado

Unmodified : Não modificado

Modified : Modificado

Staged : Pronto

Versionado : Versionado

git add →

git add \* → todos os arquivos

git commit →

git commit -m 'meu primeiro commit'

git log →

mostra os commits

## Git → Aula 2 de 4

- Snapshots:

Versionamento .git → comunicar com outros servidores → commit

- Como o histórico é armazenado no Git:

- Integridade: SHA-1 → **identificação única** → checksum (informações trafeguem de forma integra)

- Nodos de um commit:

- relacionar computacionalmente → código
- **Árvore de arquivos** → **Snapshot**

- blobs → arquivos que estão dentro do commit
- author → quem criou o repositório
- committer → quem efetuou o commit
- parent → parente do commit anterior → quem é o antecessor?
- Linhas do tempo → BRANCH:
  - Master (ou main)
  - Demais branches-
- Ponteiro HEAD
  - antes da Master (cabeça)
- Criando branches (ramificações/ linhas do tempo)
  - **git branch nomedalinhadotempo** → cria a nova linha do tempo
  - **git checkout nomedalinhadotempo** → entrar na linha do tempo → se commitar vai ser nessa linha do tempo → ponteiro head vai pra cá.
  - git show HEAD~1:./file.txt

## Git → Aula 3 e 4

- Excluir branche
  - **git branch -d nomedalinhadotempo** → deleta a linha e perde alterações
- Juntar branches (juntar ramificações/ linhas do tempo:
  - Ir para a linha principal (**git checkout nomedabbranchprincipal**)
  - **git merge nomedabbranchqueseráintroduzidanaprincipal** → não excluirá a branch secundária, que foi introduzida na principal (fica disponível para consulta)
  - Outro comando de mesclagem é **git rebase nomedabbranchqueseraintroduzidanaprincipal** → mescla e exclui a linha secundária, que foi introduzida na principal (não fica mais disponível)
- Repositório GitHub:

- **git clone linkdogithub** → clona repositório
- **git pull** → traz as informações do repositório remoto para o local
- **git push** → subir as informações do local para o remoto

Git Branch - Parte 1: <https://youtu.be/-BIXpRkqJy0>

Git Branch - Parte 2: <https://youtu.be/XsZUpiEyvgU>

Git - Repositório Remoto: <https://youtu.be/UOf8QWtlJNw>