



Manual de comandos de Git

Comandos

01

git clone → Clona um repositório já existente no github.

Sintaxe: `git clone <<url-do-repo.git>>`

02

git config → Necessário para configurar o git para uso pela primeira vez.

Sintaxe:

`git config --list` → lista as configurações do git

`git config --global user.email [fulanodetal@exemplo.b]`
`(mailto:fulanodetal@exemplo.br)r` → configura o email do usuário. Deve ser o mesmo da conta do GitHub

`git config --global user.name "Fulano de Tal"` → configura o nome do usuário.

03

git branch → Lista todas as branches presentes naquelerepositório

Sintaxe: `git branch`

💡 Digite :WQ para sair da lista que foi aberta

04

git branch <> → Cria uma nova branch com o nome escolhido

Sintaxe: `git branch exemplo-de-nome`

05

git checkout <> → entra na branch com o nome selecionado

Sintaxe: `git checkout exemplo-de-nome`

💡 Bônus: podemos combinar os dois comandos acima usando `git checkout -b exemplo-de-nome`. Dessa forma criamos a branch e a acessamos.

06

git status → lista as mudanças realizadas na branch desde o último commit

Sintaxe: `git status`

💡 Digite :WQ para sair da lista que foi aberta

07

git add → Adiciona arquivos para a **staging area**. Pode ser feito arquivo por arquivo ou adicionar a pasta inteira do projeto.

Sintaxe:

`git add -- all` → Adiciona todos os arquivos que ainda não foram adicionados

`git add .` → Adiciona todos arquivos da pasta atual

`git add nome-do-arquivo` → Adiciona o arquivo

08

git commit → Permite a criação de um novo **commit**, isto é, uma nova **etiqueta** das alterações do código em um determinado momento do tempo.

Sintaxe: `git commit -m "Descrição da alteração"`

💡 Não se esqueça das aspas aqui!

09

git push → Sobe os commits realizados para o github (ou outra plataforma de nuvem)

Sintaxe: `git push origin nome-da-branch`

10

git pull → Baixa as atualizações do repositório remoto para o repositório local.

Sintaxe: `git pull origin nome-da-branch`