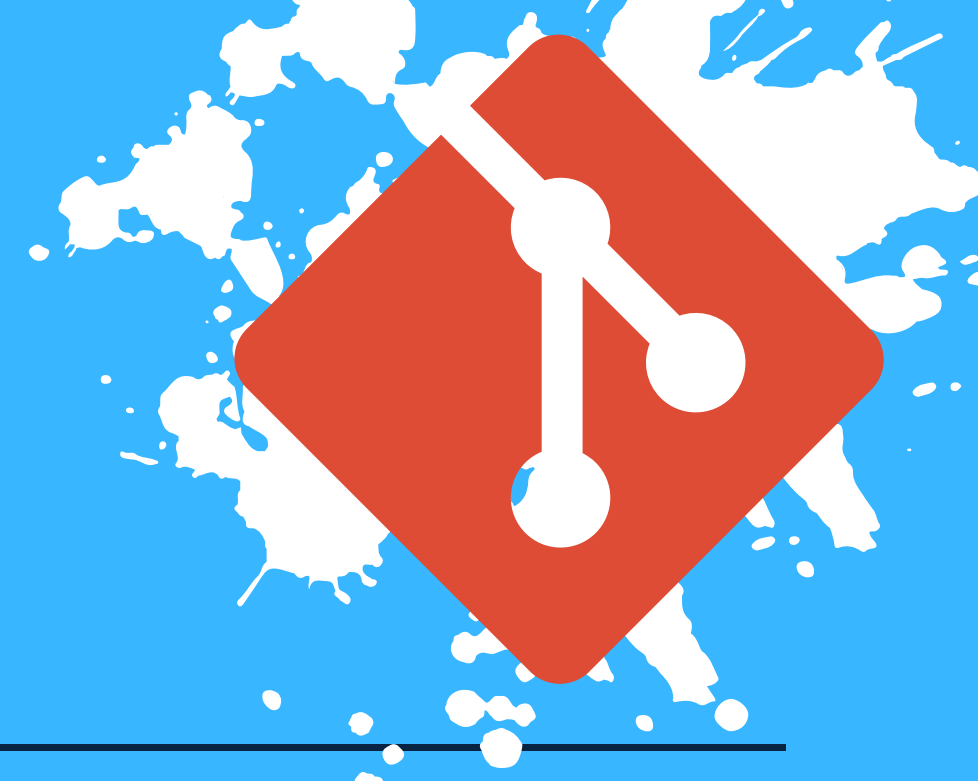


Git - Essencialidades

<https://github.com/lucaem>



Dentro da pasta local que será repositório:

1

Após tudo instalado, vá dentro da pasta de repositório, clique botão direito - Git Bash

2

O primeiro comando é o **git init** - Depois **git add** e o nome do arquivo ou **git add .** para adicionar todos os arquivos da pasta

3

Pode-se digitar **git status** para ver os arquivos a serem "commitados"

4

Para commitar é o comando **git commit -m "nome do commit"**, após pode-se verificar o status novamente.

5

Mudar o nome da principal branch de master main é **git branch -M "main"**

6

Após criar o repositório no Github, digitar no git bash **git remote add origin link**

7

Após, **git push -u origin main** e irá abrir uma janela de login autenticação do Github

Para caso primeiro uso do Git após ser instalado:

1

git config --global user.name "Seu nome aqui" git
git config --global user.email "seuemail@seudominio.com"

3

Para verificar as alterações: **git config --global --list**

4

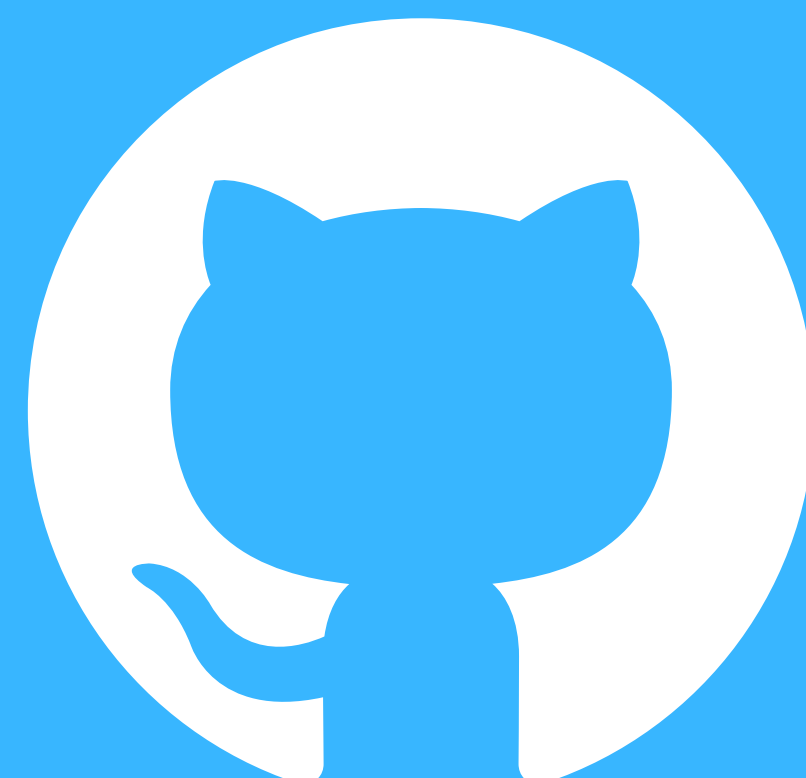
Para gerar uma chave SSH, em algum algum terminal digitar: **ssh-keygen -t rsa -C "seu comentario"**, Ele mostrará o diretório onde será salva a nossa chave e você também informará uma senha, é só ir apertando enter

5

Para visualizar a chave é:
cat ~/.ssh/id_rsa.pub - copie a chave

6

Entrar no GitHub - Settings - SSH and GPG Keys - Clicar em new SSH, dar um nome a chave e copia-la ali



<https://github.com/lucaem>

Git - Essencialidades

<https://github.com/lucaem>



Alterando e adicionando arquivo:

#1

Legal, após alterar ou adicionar um arquivo, deve-se subir as atualizações. Primeiro **git add .**, para adicionar todos os arquivos, e **git commit -m "nome do commit"**

#2

Para algo no repositório remoto, GitHub nesse caso, precisa-se dar o push, então **git push origin main** (sem o -u)

#3

O código no GitHub terá sido alterado e por lá é possível clicar no nome da commit, ver quem fez as alterações e quais foram elas.

Branch

#1

A primeira coisa que fazemos é **git checkout -b "nomedabbranch"**, esse comando além de criar a branch já entra nela com o checkout

#2

Após criar um novo arquivo ou alterar algo desejado, colocamos a nossa alteração em staging com o **git add .** e commitamos com o **git commit -m "nome da commit"**

#3

O "push" será diferente, ao invés de **git push origin main**, será **git push origin nomedabbranch**. Para voltar para a branch main é **git checkout main**, e pra voltar depois é só **git checkout nomedabbranch** novamente

Merge

#1

Para juntar as alterações da nova branch com a principal, deve-se para a branch principal com o **git checkout main**

#2

Já dentro da branch principal, com o comando **git merge nomedabbranch** junta-se tudo

#3

Para finalizar então, precisa enviar tudo para o GitHub com o **git push origin main**



<https://github.com/lucaem>

Git - Essencialidades

<https://github.com/lucaem>



Clone

#1

Sempre que entrar em um repositório, seja o seu ou o de qualquer outra pessoa, terá esse botão Code, aparecerá um link que deve ser copiado.

#2

Dentro de uma pasta, com o botão direito, abrir o Git Bash e digitar o comando **git clone linkcopiado** e tudo será "puxado" para a pasta

Pull

#1

Para puxar as atualizações no repositório de origem é só utilizar, de dentro do repositório, o comando **git pull**, ele irá puxar todas as alterações feitas no repositório do Github para o seu repositório local

Fork

#1

Você só precisa apertar nesse botão dentro do repositório e ele aparece automaticamente lá na sua conta Github

Pull request

#1

Após o fork, pode-se dar o **git add .**, **git commit -m "nomedacommit"** e **git push origin main**

#2

Dentro do GitHub irá aparecer uma mensagem do tipo: "This branch is 1 commit ahead of nomedapessoa", Isso significa que a branch do seu repositório está 1 commit "na frente" da branch original

#3

Após também irá aparecer um botão de Pull request para caso deseje enviar para o dono do repositório original uma solicitação de pull, ou seja, fazer com que ele puxe as alterações que você fez no seu repositório para o repositório dele, original

#4

Ao clicar nesse botão, você será direcionado para uma página que fará a avaliação se esse pull request terá conflitos ou não com o código no repositório original. Caso não tenha, bastão clicar no botão de Create pull request. Deve-se colocar um nome intuitivo, que demonstre a funcionalidade adicionada e o ideal é que você também crie uma boa descrição do que desenvolveu. Depois disso, basta esperar para que o dono da branch original aceite o seu pull request

<https://github.com/lucaem>