

Tutorial Passo-a-passo Git e GitHub

[Tutorial Passo-a-passo Git e GitHub](#)

[Criar chave SSH](#)

[Criando Repositório Remoto \(GitHub\)](#)

[Criando Repositório Local \(sua máquina\)](#)

[Conectar o Repositório Local ao Remoto](#)

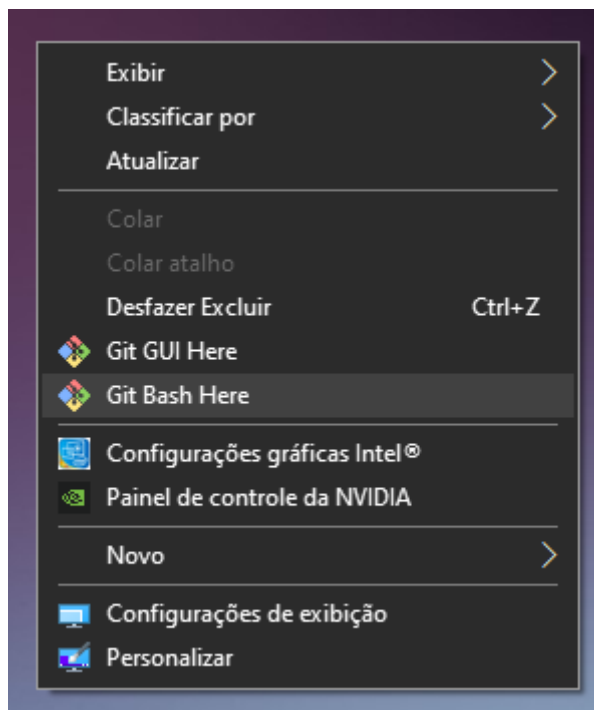
[Usando o repositório](#)

[Modificação em apenas 1 arquivo](#)

[Juntar-se à um Repositório que já existe](#)

Criar chave SSH

Para criar a chave ssh, vamos abrir o terminal do Git Bash na área de trabalho. Para isso, clique com o botão direito do mouse em um local vazio da tela e selecione a opção “Git Bash Here”, para abrir o terminal.

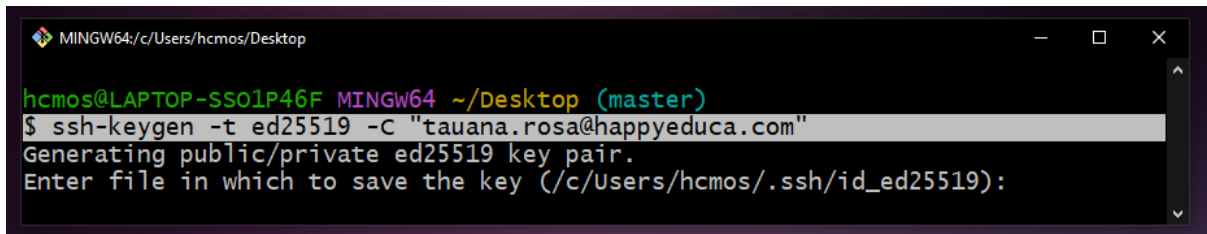


Ao abrir o terminal, digite o seguinte comando:

```
ssh-keygen -t ed25519 -C "seu.email@email.com"
```

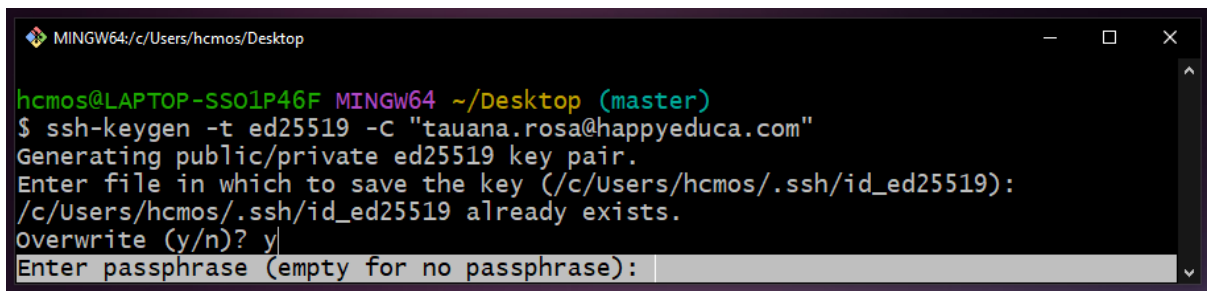
PRESTE ATENÇÃO: a letra C neste comando é maiúscula.

Após o comando, o terminal vai entrar em processo de gerar a chave. Na linha "Enter file in which to save the key" pressione enter para confirmar o caminho onde será salvo sua chave.



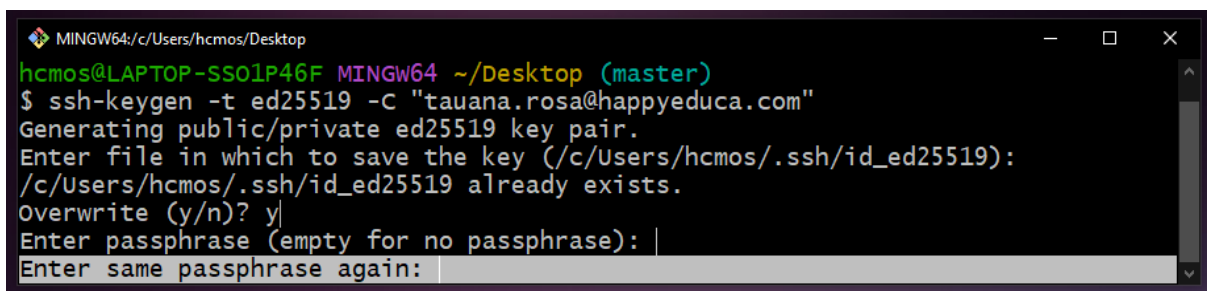
```
MINGW64:/c/Users/hcmos/Desktop
hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop (master)
$ ssh-keygen -t ed25519 -C "tauana.rosa@happyeduca.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/hcmos/.ssh/id_ed25519):
```

Após confirmar o local onde ficará a chave, insira uma senha de acesso. Se não quiser utilizar uma senha, apenas pressione enter para deixar em branco. Quando a senha é escrita, os caracteres escritos não aparecem na tela, pois é um terminal.



```
MINGW64:/c/Users/hcmos/Desktop
hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop (master)
$ ssh-keygen -t ed25519 -C "tauana.rosa@happyeduca.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/hcmos/.ssh/id_ed25519):
/c/Users/hcmos/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y|
Enter passphrase (empty for no passphrase):
```

Confirme a senha novamente ou deixe em branco.



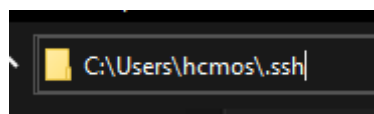
```
MINGW64:/c/Users/hcmos/Desktop
hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop (master)
$ ssh-keygen -t ed25519 -C "tauana.rosa@happyeduca.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/hcmos/.ssh/id_ed25519):
/c/Users/hcmos/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y|
Enter passphrase (empty for no passphrase): |
Enter same passphrase again:
```

A chave ssh será então gerada e estará salva no arquivo id_ed25519.pub. O caminho para este arquivo está selecionado no print abaixo.

```
MINGW64:/c/Users/hcmos/Desktop
Enter file in which to save the key (/c/Users/hcmos/.ssh/id_ed25519):
/c/Users/hcmos/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase): |
Enter same passphrase again: |
Your identification has been saved in /c/Users/hcmos/.ssh/id_ed25519
Your public key has been saved in /c/Users/hcmos/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:93w94KX28KKzAa2RRpKb/D0lAhevQj7+6QKup11MNMk tauana.rosa@happyeduca.com
The key's randomart image is:
+--[ED25519 256]--+
|      . . .      |
|      + . .      |
|    o + . .      |
|   . B = . .      |
|  +SE.+ o..o     |
| .O.O.+..+..     |
| . . * +. = .     |
| .O... =. +      |
| o+.  o*+ .o     |
+-----[SHA256]-----+

hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop (master)
$ |
```

No diretório do seu Explorador de Arquivos digite o endereço da pasta onde está o arquivo (do print anterior).



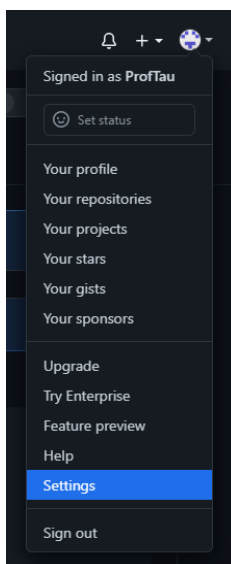
Dentro da pasta ssh, há 2 arquivos (se for a primeira vez gerando a chave) ou 4 arquivos (se for a segunda vez gerando a chave), o arquivos que iremos abrir é o com extensão “.pub”, como mostra a imagem abaixo.

Nome	Data de modificação	Tipo
id_ed25519	19/12/2022 11:05	Arquivo
id_ed25519	19/12/2022 11:05	Arquivo PUB
known_hosts	22/11/2022 14:29	Arquivo
known_hosts.old	22/11/2022 14:29	Arquivo OLD

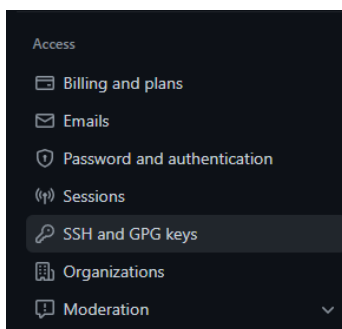
Abra o arquivo com o Bloco de Notas ou outro editor de texto simples. Não queremos editar o arquivo, apenas consultar o conteúdo para copiá-lo.



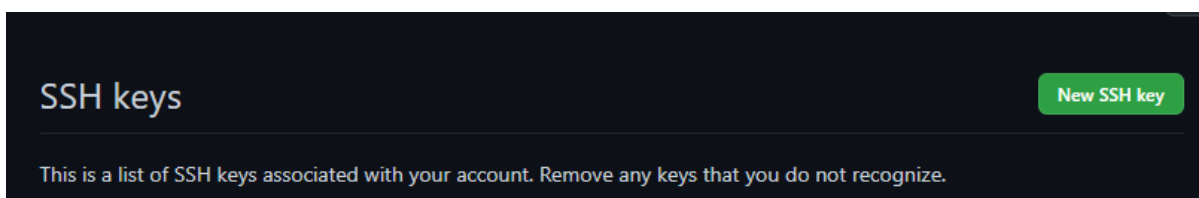
Com a chave ssh copiada, entre em seu GitHub e acesse as configurações ("settings").



Na área de configurações, entre na seção "SSH and GPG keys".



Clique na opção "New SSH key".



Preencha o formulário de cadastro da chave SSH. No título, coloque o nome do dispositivo que está usando, para identificá-lo. Na área "Key" insira a chave que copiamos do arquivo ".pub". E salve clicando no botão verde.

SSH keys / Add new

Title

meu computador pessoal

Key type


Authentication Key

Key

```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIOytZ3TnQlItqLvkWpjK6BRC03Kj3+tCTpd2xBba7rHD
tauana.rosa@happyeduca.com
```

Add SSH key

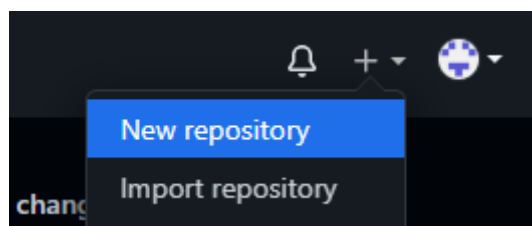
Após salvar, a lista de chaves aparecerá desta forma:

**meu computador pessoal**
SHA256:93W94KX28KKzAa2RRpKb/D01AHevQj7+6QKup11MNmk
Added on Dec 19, 2022
Never used — Read/write

Delete

Criando Repositório Remoto (GitHub)

Agora vamos criar o repositório remoto ao qual vamos nos conectar depois. No canto superior direito, clique no botão “+” e selecione a opção “New repository”.

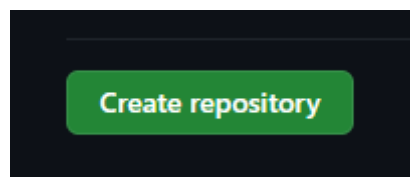


Na página de criação do repositório, preencha o formulário de criação e atente-se para marcar a visibilidade como pública.



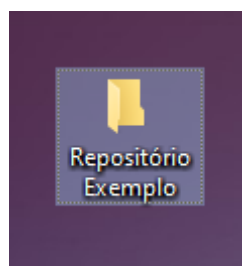
The screenshot shows the GitHub 'Create a new repository' page. At the top, it says 'Create a new repository' and provides a brief explanation of what a repository is. Below this, there are two main fields: 'Owner' and 'Repository name'. The 'Owner' field is set to 'ProfTau' with a dropdown arrow. The 'Repository name' field is set to 'ExemploGitHub' and has a green checkmark next to it. Below these fields, there is a suggestion for repository names: 'Great repository names are short and memorable. Need inspiration? How about **special-carnival**?'. There is also a 'Description (optional)' field with the text 'Tutorial de como utilizar o Git e o GitHub'. At the bottom, there are two radio buttons for visibility: 'Public' (selected) and 'Private'. The 'Public' option is described as 'Anyone on the internet can see this repository. You choose who can commit.' and the 'Private' option is described as 'You choose who can see and commit to this repository.'

Clique em “Create repository” para criar o repositório.

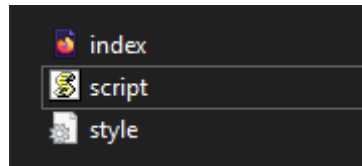


Criando Repositório Local (sua máquina)

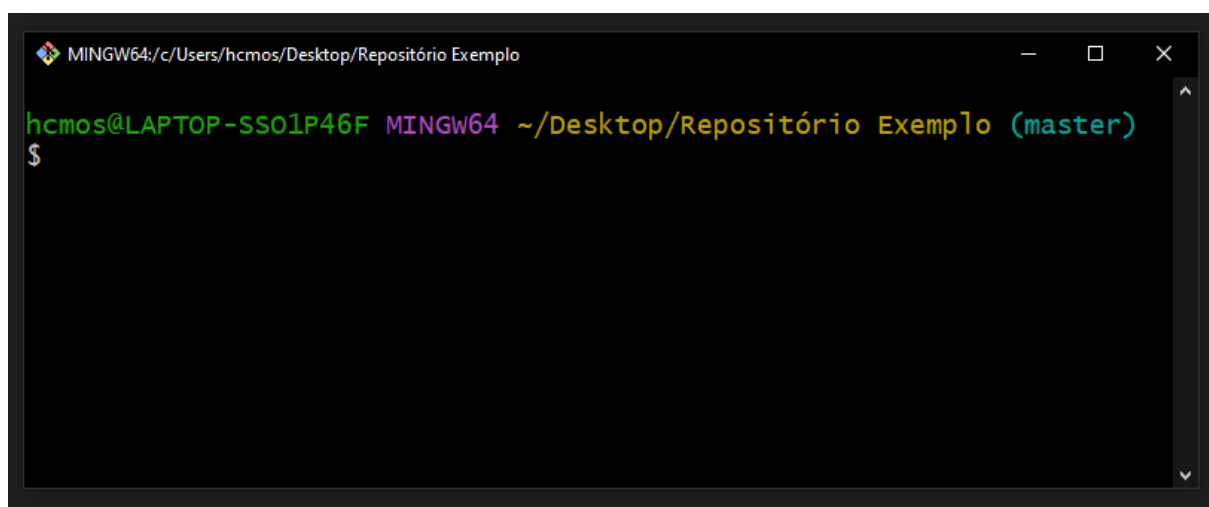
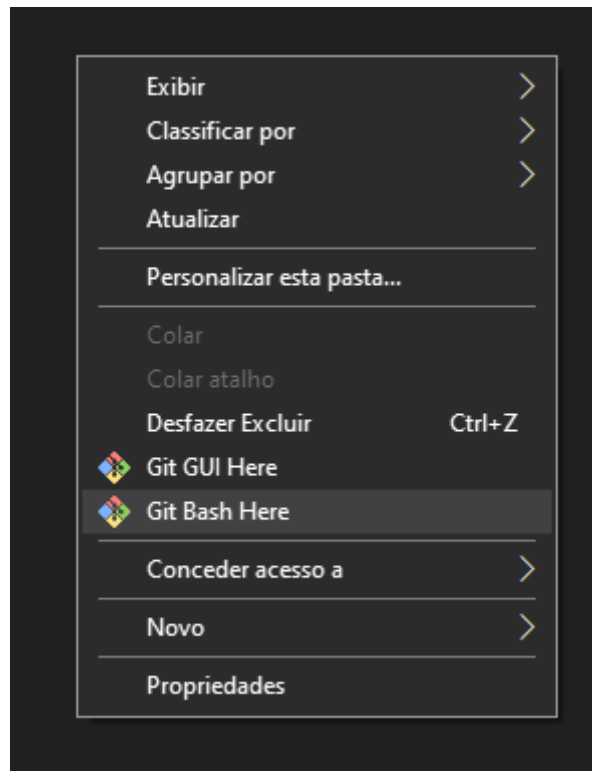
Na sua Área de Trabalho, crie uma pasta (nome à sua escolha) para ser seu repositório local.



Dentro desta pasta, adicione 3 arquivos com conteúdos diferentes.



Dentro na pasta que será o repositório, clique com o botão direito em um local vazio e selecione a opção “Git Bash Here” para abrir o terminal.



Para que esta pasta se torne um repositório de fato precisa executar o seguinte comando (destacado em branco):

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop/Repositório Exemplo (master)
$ git init
Initialized empty Git repository in C:/Users/hcmos/Desktop/Repositório Exemplo/.git/
hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop/Repositório Exemplo (master)
$ |
```

Com o repositório criado, execute o comando “git status” para ver que há arquivos pendentes.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop/Repositório Exemplo (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        index.html
        script.js
        style.css

nothing added to commit but untracked files present (use "git add" to track)
```

Para adicionar todos os arquivos pendentes, execute o seguinte comando:

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop/Repositório Exemplo (master)
$ git add .
hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop/Repositório Exemplo (master)
$ |
```

Se realizar um git status novamente, verá que os arquivos mudaram para verde, então agora estão preparados para o Commit.


```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   index.html
        new file:   script.js
        new file:   style.css

hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (master)
$
```

Para realizar o commit execute o comando destacado abaixo, não esqueça de colocar uma mensagem descritiva.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (master)
$ git commit -m "Primeiros arquivos"
[master (root-commit) 61bd3c5] Primeiros arquivos
 3 files changed, 11 insertions(+)
 create mode 100644 index.html
 create mode 100644 script.js
 create mode 100644 style.css

hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (master)
$
```

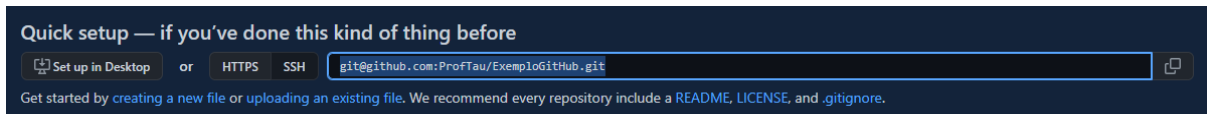
Para renomear a branch para "main" utilize o seguinte comando:

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (master)
$ git branch -m main

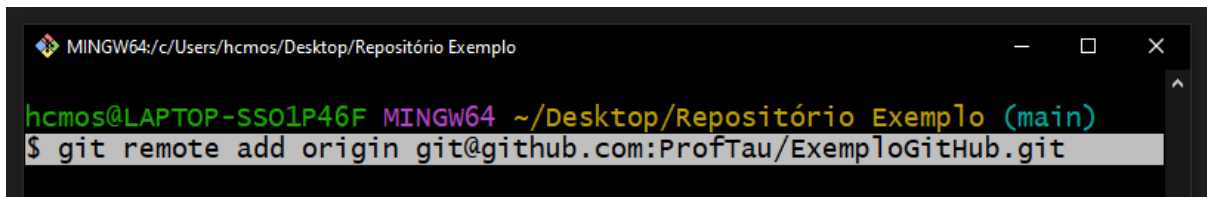
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$
```

Conectar o Repositório Local ao Remoto

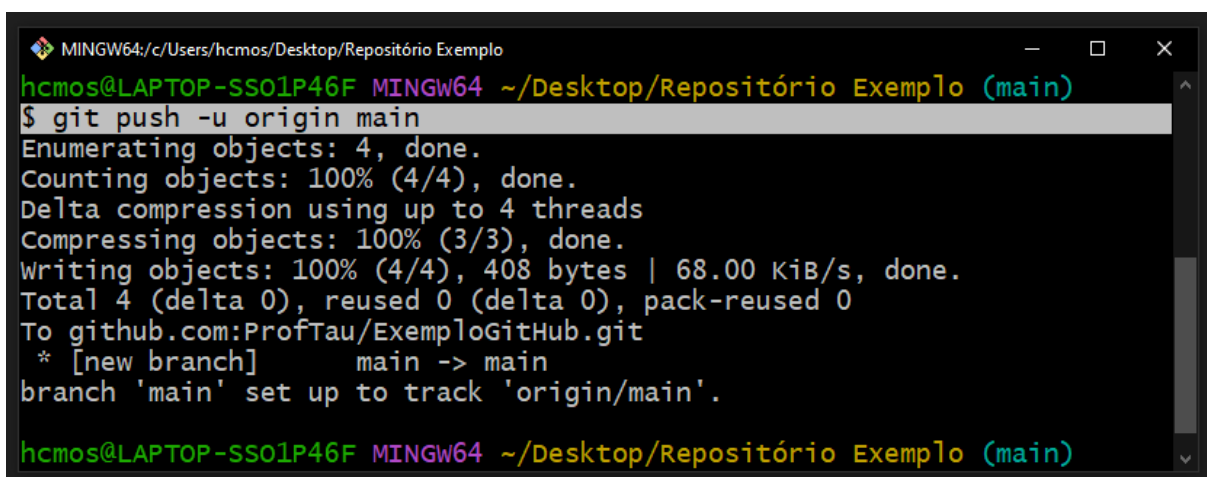
Agora que já temos a chave SSH, o repositório remoto e o local criado, podemos conectá-los. No seu repositório no GitHub, copie o endereço SSH no topo da página.



Com seu terminal Git aberto em seu repositório local, execute o comando “git remote add origin <insira aqui o endereço SSH copiado do GitHub>”.

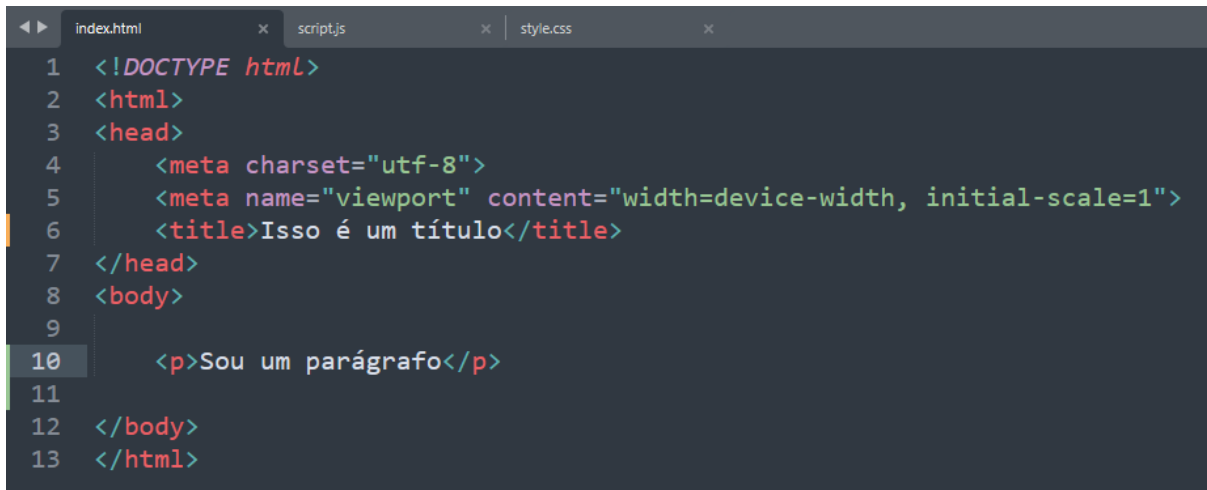


Para enviar seus commits ao repositório remoto, utilize o seguinte comando:



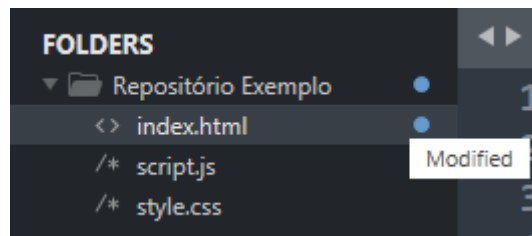
Usando o repositório

Vamos abrir o documento HTML e adicionar conteúdo em algumas linhas, para que ele sofra modificações.

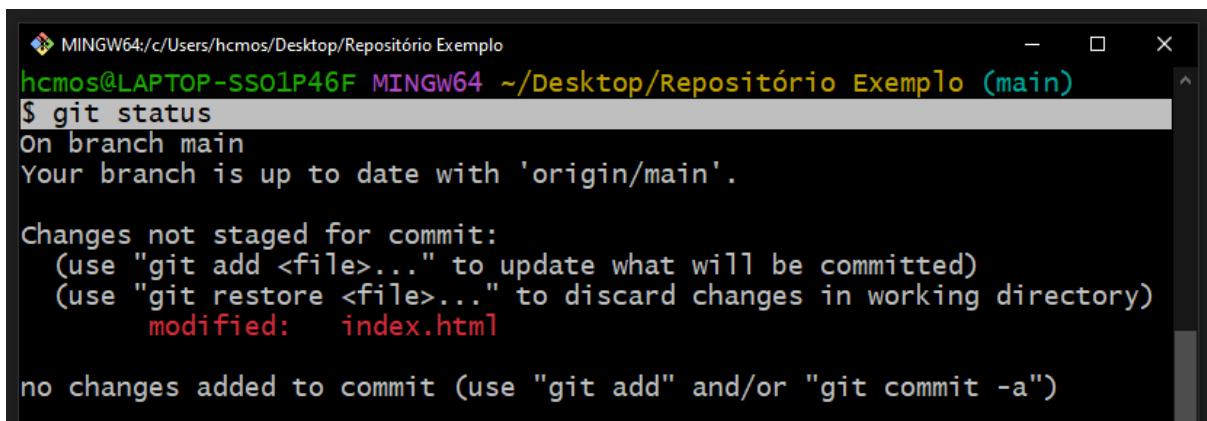


```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title>Isso é um título</title>
7 </head>
8 <body>
9
10  <p>Sou um parágrafo</p>
11
12 </body>
13 </html>
```

Note que, se você utilizar o Sublime, ele reconhece que esta pasta é um repositório e indica com bolinhas azuis quando há modificações nos arquivos.



No terminal do Git, execute o `git status`, verá que apenas o arquivo HTML está com o status de modificado.

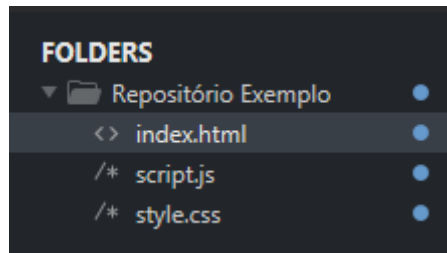


```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
```

Modifique os outros dois arquivos e salve.



Realize novamente o git status, verá que agora os 3 arquivos estão modificados. Para poder realizar o commit é necessário adicionar todas as modificações.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   index.html
        modified:   script.js
        modified:   style.css

no changes added to commit (use "git add" and/or "git commit -a")
```

Para adicionar somente um arquivo utilize o comando “git add <nome do arquivo>”.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git add index.html
```

Realizando o git status novamente, verá que o arquivo HTML estará pronto para commit, mas os demais arquivos não foram adicionados, por isso ainda estão com status modificado.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   index.html

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   script.js
        modified:   style.css
```

Para adicionar todos os arquivos de uma vez, sem nomeá-los utilize o comando “git add .”

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git add .
```

Se realizar o git status novamente verá que todos os arquivos estão prontos para o commit.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   index.html
        modified:   script.js
        modified:   style.css
```

Realize o commit com o comando *git commit -m “sua mensagem”*

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git commit -m "html com conteúdo e outros com comentário"
[main e81a1a3] html com conteúdo e outros com comentário
3 files changed, 5 insertions(+), 1 deletion(-)
```





Se realizar o git status novamente, verá que desta vez não há mais nada na fila para ser adicionado ou committado.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

Se realizarmos o comando git push agora, os arquivos serão sincronizados com o repositório remoto.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git push
Enumerating objects: 9, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 559 bytes | 559.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:ProfTau/ExemploGitHub.git
 61bd3c5..e81a1a3  main -> main
```

	ProfTau html com conteúdo e outros com comentário	e81a1a3 2 minutes ago	🕒 2 commits
	index.html	html com conteúdo e outros com comentário	2 minutes ago
	script.js	html com conteúdo e outros com comentário	2 minutes ago
	style.css	html com conteúdo e outros com comentário	2 minutes ago

Modificação em apenas 1 arquivo

Se eu quiser modificar apenas o arquivo HTML, não preciso commitar os outros arquivos junto.

```
<p>Sou um parágrafo</p>
<p>apenas uma modificação</p>
```

Realize o git status e verá que há apenas um arquivo modificado.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
```

Adicione o arquivo HTML.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git add index.html
```

Com git status verá que foi adicionado.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   index.html
```





Realize o commit descrevendo qual foi a modificação.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git commit -m "apenas html atualizado"
[main 30c85f4] apenas html atualizado
1 file changed, 1 insertion(+)
```

Realize o push.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 399 bytes | 26.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:ProfTau/ExemploGitHub.git
    e81a1a3..30c85f4  main -> main
```

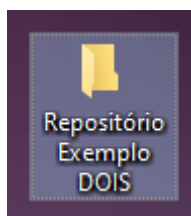
Ao consultar seu repositório no Github, verá que apenas o arquivo HTML está com a descrição do último commit realizado.

	ProfTau apenas html atualizado	30c85f4 1 minute ago	🕒 3 commits
	index.html	apenas html atualizado	1 minute ago
	script.js	html com conteúdo e outros com comentário	7 minutes ago
	style.css	html com conteúdo e outros com comentário	7 minutes ago

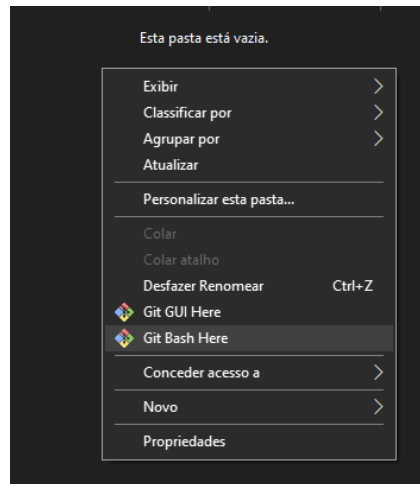
Juntar-se à um Repositório que já existe

Criei outra pasta na Área de Trabalho para simular outra pessoa tentando se conectar ao repositório.

Se for alguém em outra máquina, precisará criar a chave SSH e conectar ao GitHub.



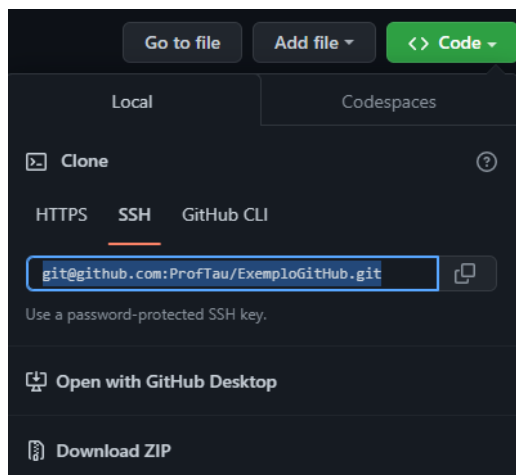
Dentro da pasta, abra o GitBash.



Execute o comando para tornar a pasta um repositório.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo DOIS
hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop/Repositório Exemplo DOIS (master)
$ git init
Initialized empty Git repository in C:/Users/hcmos/Desktop/Repositório Exemplo DOIS/.git/
```

No repositório do GitHub, clique na opção “Code” e copie o endereço SSH.






Conecte remotamente, usando o endereço SSH copiado.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo DOIS
hcmos@LAPTOP-SSO1P46F MINGW64 ~/Desktop/Repositório Exemplo DOIS (master)
$ git remote add origin git@github.com:ProfTau/ExemploGitHub.git
```

Execute o comando “git pull origin main” para “puxar” os arquivos do repositório remoto para o seu local.

```
MINGW64:/c/Users/hcmos/Desktop/Repositório Exemplo DOIS
hcmos@LAPTOP-SS01P46F MINGW64 ~/Desktop/Repositório Exemplo DOIS (master)
$ git pull origin main
From github.com:ProfTau/ExemploGitHub
* branch          main          -> FETCH_HEAD
```

Se abrir a pasta do repositório, verá que os arquivos terão sido criados lá.

	index	19/12/2022 11:51	Firefox HTML Document	1 KB
	script	19/12/2022 11:51	Arquivo JavaScript	1 KB
	style	19/12/2022 11:51	Documento de folha de estilos em cascata	1 KB