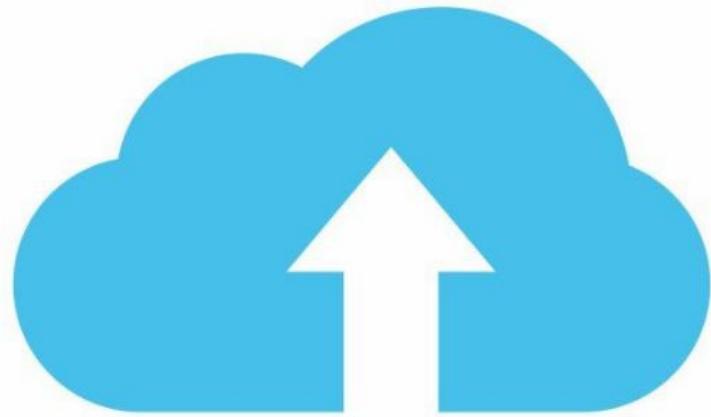
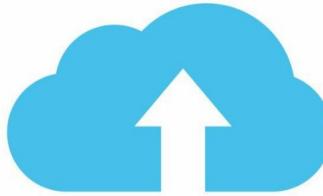


Criando repositório na interface WEB e clonando localmente



<https://github.com/>

Criando um repositório no GitHub



Criar repositório na interface WEB do GitHub

Acessar o site e logar com a conta já criada: <https://github.com/>

Clique em **Repositories > New**

Forneça um nome.

Escolha a visibilidade, publica ou privada.

Marque a opção criar README file (opcional).

Clique em **Create repository**.

Criando um repositório no GitHub



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?

[Import a repository](#).

Required fields are marked with an asterisk (*).

Owner *

Repository name *

/ repo2

Great repository names are short and memorable. Need inspiration? How about [stunning-enigma](#) ?

Description (optional)

Public

Anyone on the internet can see this repository. You choose who can commit.

Private

You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file

This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses](#).

This will set `main` as the default branch. Change the default name in your [settings](#).

i You are creating a private repository in your personal account.

[Create repository](#)

Criando um repositório no GitHub



repo1 Private

Unwatch 1 Fork 0 Star 0

main 1 Branch 0 Tags Go to file Add file Code About

Initial commit f4c97eb · now 1 Commit

README.md Initial commit now

README

repo1

No description, website, or topics provided

Readme Activity 0 stars 1 watching 0 forks

Releases No releases published Create a new release

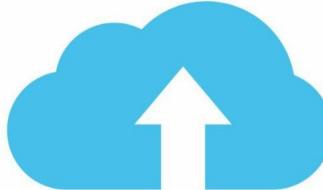
Packages No packages published Publish your first package

A screenshot of a GitHub repository page for "repo1". The repository is private. It shows one branch ("main"), one tag, and one commit. The commit is titled "Initial commit" and was made by "f4c97eb" at "now". The commit message is "1 Commit". There is a file named "README.md" with the same commit details. Below the file listing is a section for the "README" file, which contains the text "repo1". To the right of the repository details, there are sections for "About", "Releases", and "Packages", each showing zero items. At the bottom, there is a footer with links to GitHub's terms, privacy, security, status, docs, contact, and cookie management, along with a "Do not share my personal information" checkbox.



© 2025 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

Criando um repositório no GitHub



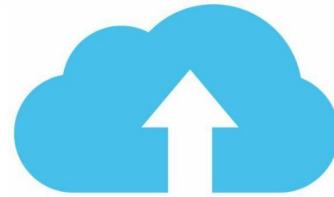
Agora com o repositório criado, podemos clonar para começar a trabalhar localmente.

Vamos usar a opção SSH.

Clicar no botão Code > SSH

Copiar o link do repositório.

Criando um repositório no GitHub



repo1 Private

Unwatch 1 Fork 0 Star 0

main 1 Branch 0 Tags

Initial commit

README.md Initial commit

README

repo1

About

No description, website, or topics provided.

Clone

HTTPS SSH GitHub CLI

git@github.com: :/repo1.git

Use a password-protected SSH key.

Readme

Activity

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

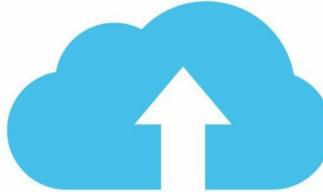
[Publish your first package](#)

The screenshot shows a GitHub repository page for "repo1". A red box highlights the "Code" dropdown menu, which is open to show cloning options via HTTPS, SSH, and GitHub CLI. The SSH URL is displayed as "git@github.com: :/repo1.git". Below the URL, a note says "Use a password-protected SSH key." To the right of the code menu, there's an "About" section with basic repository statistics and links to "Readme", "Activity", and "Releases". At the bottom, there's a "Packages" section and a note about publishing packages.



© 2025 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact](#) [Manage cookies](#) [Do not share my personal information](#)

Clonando um repositório no GitHub



Abrir o Git Bash.

Navegar até a pasta onde se deseja trabalhar.

Usar o comando `git clone` para:

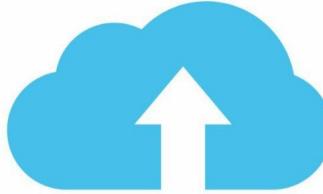
- Clonar o repositório remoto na pasta local.



```
$ cd Documents/  
  
$ git clone git@github.com:fulano/repo1.git
```

Forneça a senha quando solicitado.

Clonando um repositório no GitHub



O conteúdo (arquivos, pastas e subpastas) será baixado para o repositório local.

Usar os comandos de navegação para acessar a pasta.

Também pode ser acessado via ambiente gráfico.



```
$ ls  
'Meus Vídeos'@      'Minhas Imagens'@  
'Minhas Músicas'@    repo1/
```

```
$ cd repo1/
```

```
$ ls
```

```
README.md
```



**Pronto já podemos começar a
trabalhar com nosso repositório.**



Criando repositório local
e sincronizando com GitHub



Criar e sincronizar repositório local

Dados do usuário local



Configurações iniciais do usuário do GIT local.

Quando um commit é realizado, é gerado um log com informações do usuário que realizou o commit.

Esses dados do usuário devem ser configurados também localmente (via Git Bash)

```
$ git config --global user.name "funalo"  
$ git config --global user.email "funalo@exemplo.com"
```

Criar e sincronizar repositório local

Criando repositório local



Existe a possibilidade de criar um repositório/pasta local e posteriormente sincronizar com o GitHub.

Criar a pasta: *pode ser feito via ambiente gráfico ou linha de comando.*

Seguem comandos via Git Bash:

\$ pwd	#visualizar pasta atual
/c/Users/fulano	
\$ cd Documents/	#acessar pasta Documentos
\$ mkdir projeto1	#criar pasta para o repositório
\$ cd projeto1	#acessar pasta do repositório
\$ echo "Arquivo inicial do projeto1" >> principal.txt	#criar aquivo inicial com uma fraze
\$ ls	#listar conteúdo da pasta
principal.txt	

Criar e sincronizar repositório local

Criando repositório local



Após criar a pasta e seu conteúdo inicial é necessário inicializar o repositório para que o GIT consiga rastrear a pasta e as alterações em seu conteúdo.

Isso é feito com o comando `git init`.

```
$ pwd  
/c/Users/fulano/Documents/projeto1  
  
$ git init  
Initialized empty Git repository in C:/Users/fulano/Documents/projeto1/.git/
```

Com o repositório inicializado agora podemos adicionar os arquivos e commitar.

```
$ git status  
  
$ git add .  
  
$ git commit -m "Commit inicial do projeto1"
```

Criar e sincronizar repositório local

Sincronizando repositório local

Com o repositório local inicializado, agora para sincronizar com o GitHub, é necessário criar um repositório remoto vazio (sem arquivo README) com o mesmo nome do repositório local.

NÃO selecionar

Isso criará um repositório vazio.
Agora podemos fazer o upload.



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner *



Repository name *

projeto1

✓ projeto1 is available.

Great repository names are short and memorable. Need inspiration? How about [probable-rotary-phone](#) ?

Description (optional)

Public

Anyone on the internet can see this repository. You choose who can commit.

Private

You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

 You are creating a private repository in your personal account.

Create repository

Criar e sincronizar repositório local

Sincronizando repositório local



Fluxo para sincronizar o repositório local com o remoto.

```
$ echo "Projeto1 GIT" >> README.md          #criando arquivo README  
$ git add .                                     #adicionar arquivos  
$ git commit -m "Add arquivo README"           #commitar arquivos  
$ git branch -M main                           #ajustar nome da branch principal  
$ git remote add origin git@github.com:fulano/projeto1.git #vincular local e remoto  
$ git push --set-upstream origin main           #upload do conteúdo local
```



**Pronto já podemos começar a
trabalhar com nosso repositório.**

