

Em nossas aulas e trabalhos vamos utilizar o GitHub para versionar e hospedar nossos códigos.



<https://github.com/>

Criando conta no GitHub

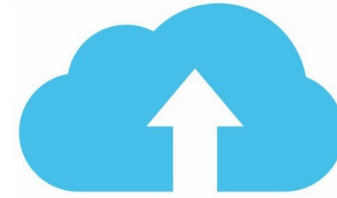


Para criarmos nossos repositórios no GitHub primeiramente temos que criar uma conta e configurar o acesso.

Quando o repositório é público, qualquer pessoa, mesmo sem conta, pode baixar o conteúdo do repositório.

Porém, para enviar algum conteúdo para o repositório é necessário ter uma conta e passar por uma autenticação. Mesmo que o repositório seja seu.

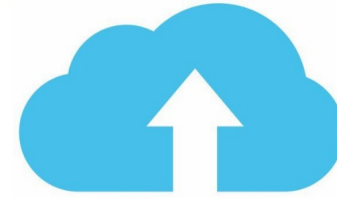
Criando conta no GitHub



Acessar o site: <https://github.com/>

- Clicar em sign up.
- Passe pelo processo de cadastro.
- Anote seu nome de usuário, ele será necessário.
- Um repositório remoto pode ser acessado pelo git local usando HTTPS ou SSH.
- No caso de HTTPS ele exigirá a criação de TOKEN que deve ser usado sempre que for realizar uma ação.
- No caso do SSH ele pedirá somente a senha da chave SSH que deve estar cadastrada no GitHub. Tornando o processo mais simples no dia a dia.

Clonando com HTTPS



Usando HTTPS

Antes de clonar um repositório via HTTPS, é necessário criar o token de acesso.

Os tokens do GitHub são baseados em permissões de acesso.

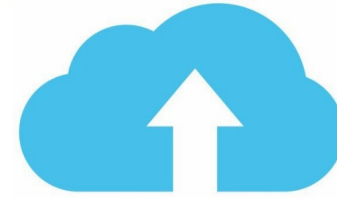
Para realizar as operações básicas:

- Clonar `git clone`
- Buscar atualizações `git pull`
- Enviar alterações `git push`

As permissões mínimas necessárias são:

- Contents - Read and Write
- Metadata - Read

Clonando com HTTPS



Gerando e configurando o token

Acesse:

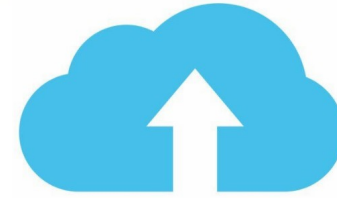
- Settings > Developer Settings > Personal Access Tokens;
- Clique em **Generate new token (fine-grained)**;
- Escolha o **repositório específico**;
- Em **Repository permissions**, defina:
 - **Contents: Read and Write**;
 - **Metadata**: será ativada automaticamente como **Read-Only**.

O token, que deve ser copiado e guardado.

Quando sair da página não será possível ver novamente o valor do token.

Tokens podem ser apagados e criados quando for necessário.

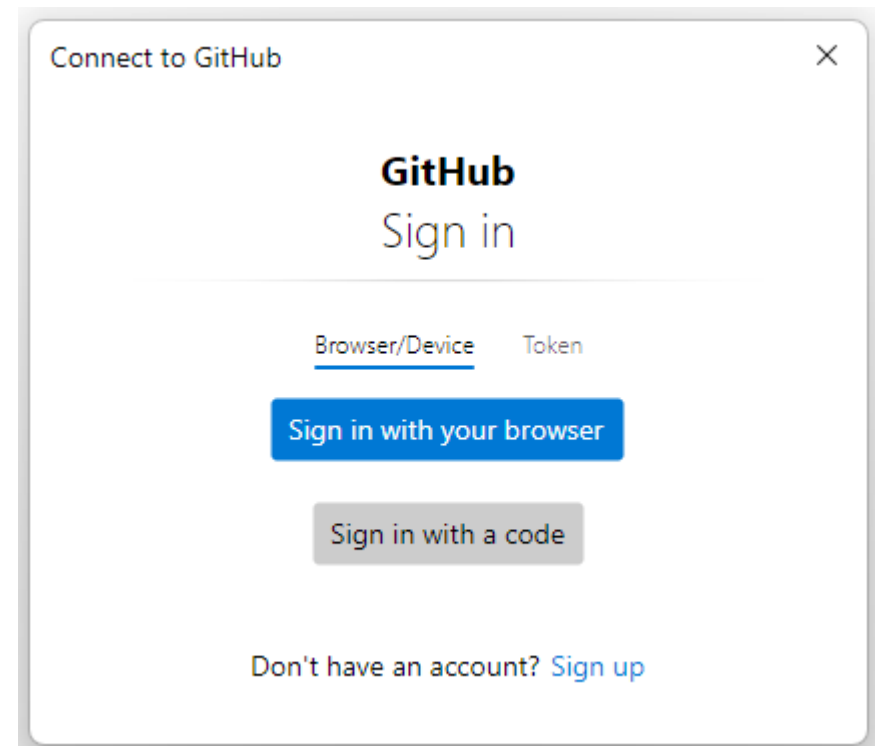
Clonando com HTTPS



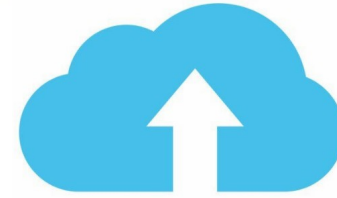
Autenticação

No caso de links HTTPS, momento `git clone` ou, do primeiro `push` ou `pull`, o sistema exibirá uma janela com dois métodos de autenticação.

- Pelo browser:
 - Pelo browser, o acesso será permitido a todos os repositórios da conta utilizada.
- Pelo token
 - Pelo token, o acesso será somente nos repositórios que o token foi configurado para permitir.



Clonando com HTTPS



Autenticação

Após a conexão o Windows armazena as credenciais utilizadas e no próximo acesso, não será necessário se autenticar novamente.

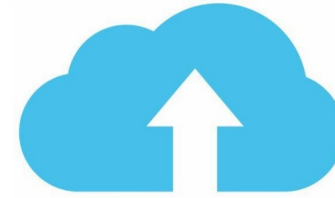
Nem sempre esse comportamento é desejado, no caso de acesso via máquinas públicas (utilizadas por vários usuários) por exemplo.

Nestes casos, para proteger seus repositórios, o usuário deve remover as credenciais do gerenciador do Windows.

Quando precisar acessar novamente, a janela de autenticação irá aparecer.

O novo acesso pode ser realizado normalmente.

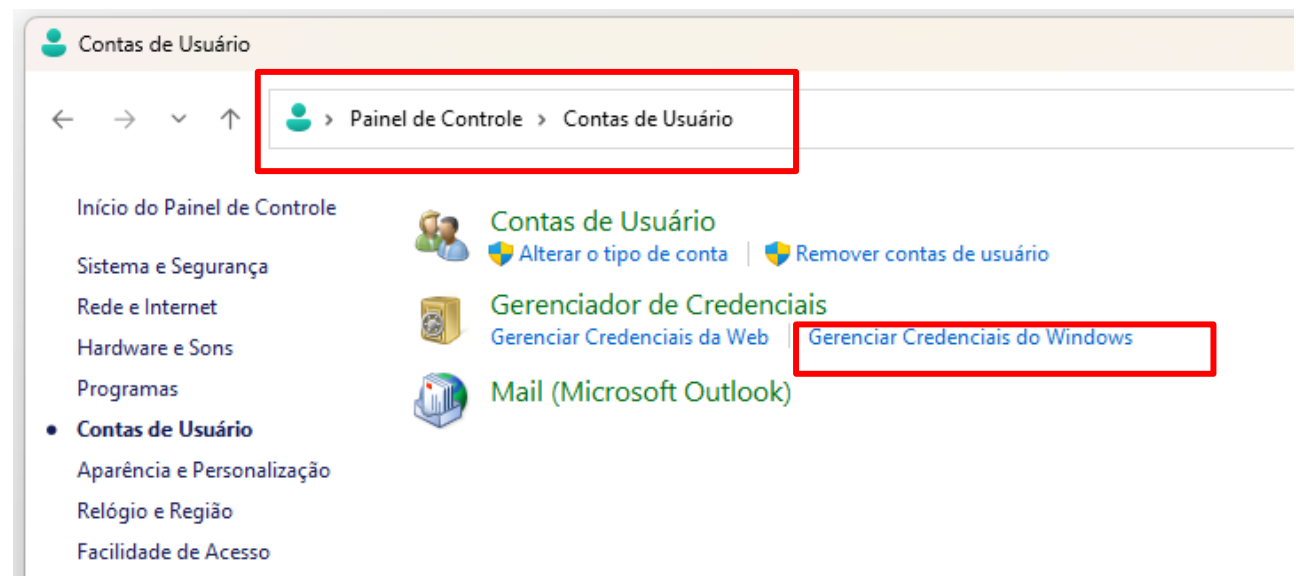
Clonando com HTTPS



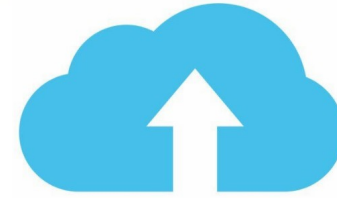
Autenticação

Removendo credenciais salvas no Windows:

- Acesse o **Painel de Controle > Contas de Usuários**
- Em **Gerenciador de Credenciais**
- Acesse **Gerenciar Credenciais do Windows**



Clonando com HTTPS



Autenticação

Removendo credenciais salvas no Windows:

- Selecione **Credenciais do Windows**
- Na lista, procure pelas credenciais *git:*<https://github.com>
- Clique na credencial e selecione **Remover**

Obs.: Imagem na próxima tela.

Clonando com HTTPS





Gerenciador de Credenciais

← → ▾ ▴ > Painel de Controle > Contas de Usuário > Gerenciador de Credenciais

Gerencie suas credenciais

Início do Painel de Controle Exibir e excluir suas informações de logon salvas para sites, aplicativos conectados e redes.

 **Credenciais da Web**

 **Credenciais do Windows**

[Fazer Backup das Credenciais](#) [Restaurar Credenciais](#)

Credenciais do Windows [Adicionar uma credencial do Windows](#)

Sem credenciais do Windows.

Credenciais Baseadas em Certificado [Adicionar uma credencial baseada em certificado](#)

Nenhum certificado.

Credenciais Genéricas [Adicionar uma credencial genérica](#)

DriveFS_103376997651857870147 Modificado: Hoje ▾

git:https://github.com Modificado: Hoje ▴

Endereço de rede ou na Internet: git:https://github.com

Nome de usuário: [REDACTED]

Senha: ooooooooo

Persistência: Computador local

[Editar](#) [Remover](#)

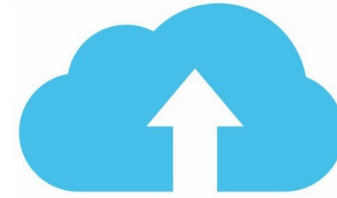
Consulte também

Contas de Usuário

Microsoft_OneDrive_Cookies_v2_Business1_https://un... Modificado: 08/07/2025 ▾

MicrosoftAccount:user=hyde x@msn.com Modificado: Hoje ▾

Clonando com HTTPS

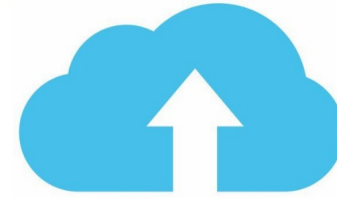


Clonando via terminal.

- Acesse a pasta onde será clonado o repositório;
- Usar o comando `git clone url_do_repositório`;
- Caso o repositório seja privado, será necessário o uso do nome de usuário e um de token de acesso no lugar da senha;
- Se tudo estiver correto, o conteúdo do repositório será baixado para a pasta local.

```
git clone https://github.com/fulano/repo1.git
Username for 'https://github.com': fulano
Password for 'https://fulano@github.com': colar_o_token_criado
```

Criando Chaves SSH



Usando SSH

O GitHub disponibiliza uma documentação para auxiliar no acesso via SSH

<https://docs.github.com/pt/authentication/connecting-to-github-with-ssh>

Caso o usuário já possua chaves, pode utilizar.

Caso não possua, ele deve criar.

As chaves podem ser criadas via Git Bash instalado na máquina Windows junto ao GIT.

O processo está descrito no site:

<https://docs.github.com/pt/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent>

Criando Chaves SSH



Abra o Git Bash

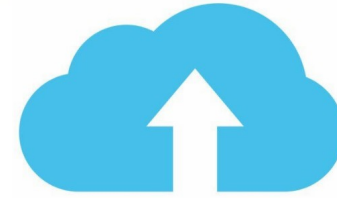
Use o comando, substituindo o e-mail pelo seu endereço de e-mail do GitHub.

```
ssh-keygen -t ed25519 -C "your_email@example.com"  
> Generating public/private ALGORITHM key pair.
```

Quando solicitado, forneça no nome do arquivo e observe onde está sendo salvo.

```
> Enter file in which to save the key  
(/c/Users/YOU/.ssh/id_ALGORITHM):chave_git_fulano
```

Criando Chaves SSH



Será solicitada uma senha para usar a chave.

```
> Enter passphrase (empty for no passphrase): *****  
> Enter same passphrase again: *****
```

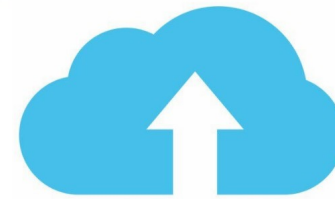
Dessa forma foram criadas duas chaves, a privada e pública.

```
> ls  
chave_git_fulano  
chave_git_fulano.pub
```

A chave pública deve ser adicionada no GitHub. (arquivo com extensão .pub)

A chave privada deve ser mantida na máquina.

Configurando Chaves SSH



Adicionando a chave pública no GitHub.

Visualizar a chave.

```
$ cat chave_git_fulano.pub  
ssh-ed25519 AAFAC3NzaC1lZDI1NTE5ACCAIMdq78ACHiZ9FsbVjmUR5XWjkDXXL7kD5fJmSlZ6S0go  
your_email@example.com
```

Copie o **conteúdo** do arquivo .pub.

No GitHub, navegue até Settings > SSH and GPG Keys > SSH Keys > New SSH key.

Forneça um nome para a chave e cole o conteúdo copiado no campo Key.

Configurando Chaves SSH



O SSH do Git Bash, procura por chaves privadas com nomes específicos.

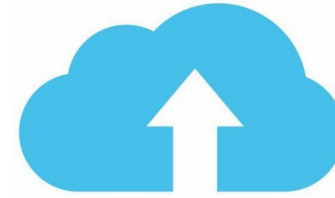
Caso você tenha personalizado o nome da sua chave, para evitar problemas, você pode fazer uma cópia da chave privada, com um nome que o SSH reconhecerá.

Exemplo:

De **chave_git_fulano** para **id_rsa**

```
$ cd ~/.ssh/           #acessar pasta de chaves
$ ls                   #listar as chaves
chave_git_fulano
chave_git_fulano.pub
$ cp chave_git_fulano id_rsa      #copia o arquivo com outro nome
$ls
chave_git_fulano          id_rsa
chave_git_fulano.pub
```


Configurando Chaves SSH



Adicionalmente, caso queira usar uma chave com nome específico:

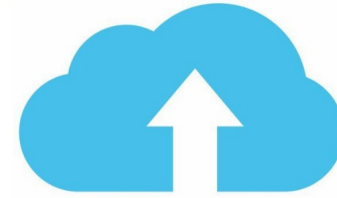
Quando for clonar um repositório via SSH deverá indicar o caminho para a chave privada (arquivo que não tem extensão).

```
$ git clone git@github.com:fulano/repo1.git --config  
core.sshCommand="ssh -i ~/.ssh/chave_git_fulano"
```

Existem várias formas de se trabalhar com chaves SSH, isso pode variar de um Sistema Operacional para outro.

Esse procedimento é para Windows quando **NÃO** se tem permissões de administrador. Nesse caso é necessário indicar a chave sempre que clonar um novo repositório.

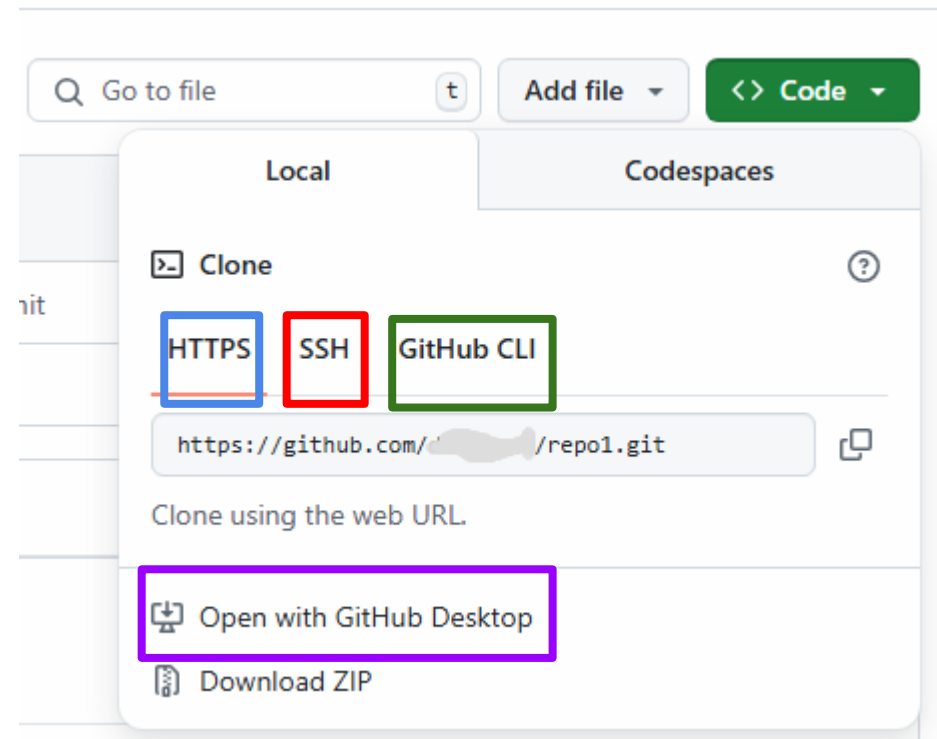
Acesso ao repositório GitHub



O GitHub provê outras formas de clonar os repositórios.

Você pode explorar, e descobrir uma em que se adapte melhor.

- HTTPS
- SSH
- GitHub CLI
- GitHub Desktop



**Agora podemos baixar e enviar
conteúdo para nosso repositório.**

