



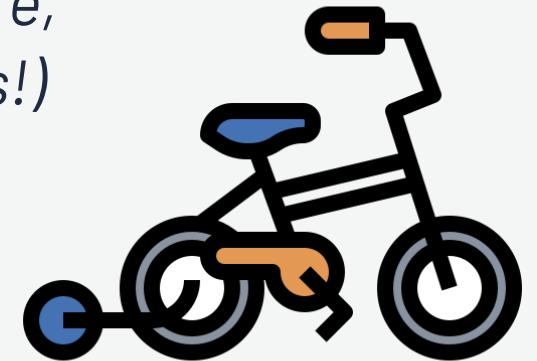
SÃO
PAULO
TECH
SCHOOL

Pesquisa e Inovação

Git e GitHub

Training Wheels

("rodinha de bicicleta", ou seja, te ajuda no começo e, depois de praticar bastante, você não precisa mais!)



Qual ferramenta usar de acordo com seu sistema operacional:



Git bash

Para executar comandos git no **Windows**, baixe o **GitBash**:

<https://git-scm.com/downloads>



Terminal

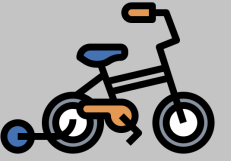
O atalho **Linux** para abrir o terminal é: **Ctrl+Alt+T**.



Terminal



No **MacOS**, abra o Spotlight clicando em **Command + Espaço**, (ou clique no ícone do Launchpad no Dock) **digite Terminal** no campo de busca e clique em Terminal.



Training wheels

Antes de executar os comandos git aprendidos em aula (git clone, git pull, git add, git commit, git push, git status...) você precisa:

- Limpar as credenciais do usuário anterior que usou a máquina
 - (não necessário quando você está usando sua máquina pessoal)
- Adicionar suas credenciais
- Gerar seu token de acesso
- Acessar o diretório/pasta onde colocará seu projeto

Durante a execução dos comandos git aprendidos em aula, você deve:

- Seguir os passos com atenção

Depois de executar os comandos git aprendidos em aula, você precisa:

- Limpar suas credenciais
- Apagar o diretório/pasta onde colocou seu projeto

Training wheels

Para efetuar os comandos que apresentaremos neste tutorial e nas aulas, abra o GitBash (ou Terminal) e aguarde o **cursor** piscar, indicando que está aguardando seu comando.
Digite o comando desejado e então tecla **[Enter]** no seu **teclado**.



```
MINGW64:/c/Users/Fernanda Caranico
Fernanda Caranico@CARANICO: ~$ |
```

Este texto em amarelo indica "onde" do seu computador você está.
O til ~ indica que estamos no diretório (pasta) raiz do usuário, ou seja, em:
C:\Users\SeuUsuario
Também Podemos ver o caminho no título da janela:
c/Users/SeuUsuario

Antes dos comandos Git:

Como limpar as credenciais do
usuário anterior que usou a
máquina

Training wheels

Limpe as credenciais → para efetuar os comandos abaixo, utilize a ferramenta de acordo com seu sistema operacional, que poderá ser GitBash, Terminal Linux ou Terminal Mac.

```
git config --global --unset user.name
```

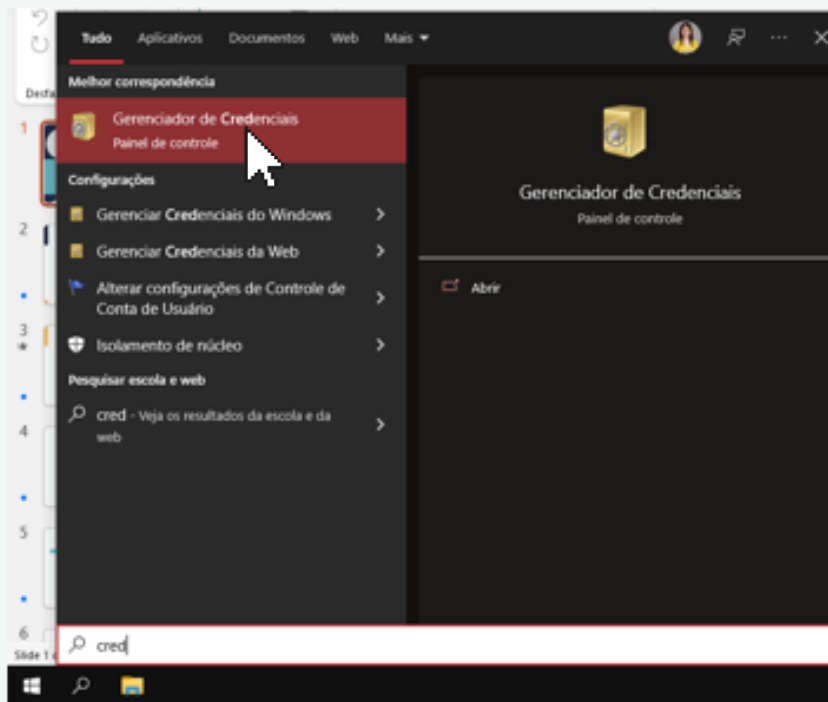
```
git config --global --unset user.email
```

Ao efetuar os comandos, parecerá que nada aconteceu, mas pode prosseguir.

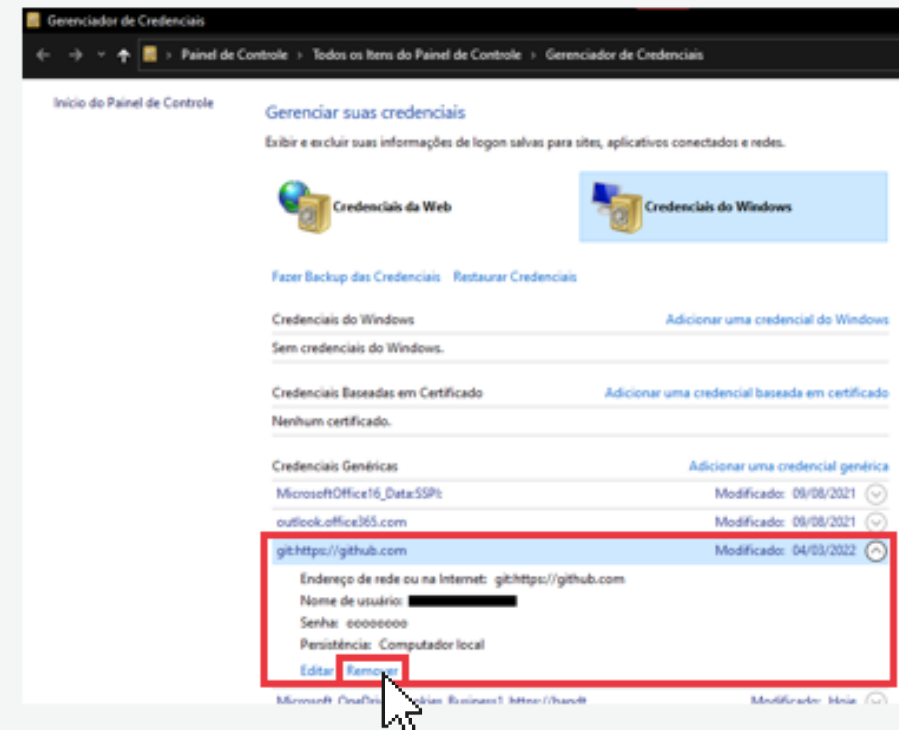
Training wheels

No Windows, verifique se há algum registro em Gerenciador de Credenciais e remova:

1. Busque por "Gerenciador de Credenciais"



2. Identifique a credencial "github" e clique em "Remover"



Antes dos comandos Git:

Como adicionar as suas
credenciais

Training wheels

Insira as suas credenciais de acesso

```
git config --global user.name SeuNomeDeUsuario
```

```
git config --global user.email seu@email.com
```

Verifique que suas alterações foram efetuadas com sucesso:

```
git config --global user.name
```

```
- Aqui deve ser exibido o nome que você inseriu acima
```

```
git config --global user.email
```

```
- Aqui deve ser exibido o e-mail que você inseriu acima
```

Antes dos comandos Git:

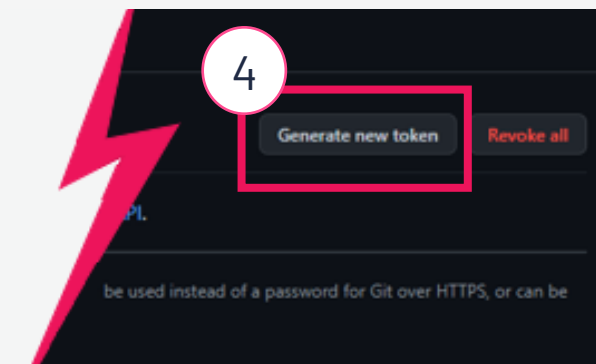
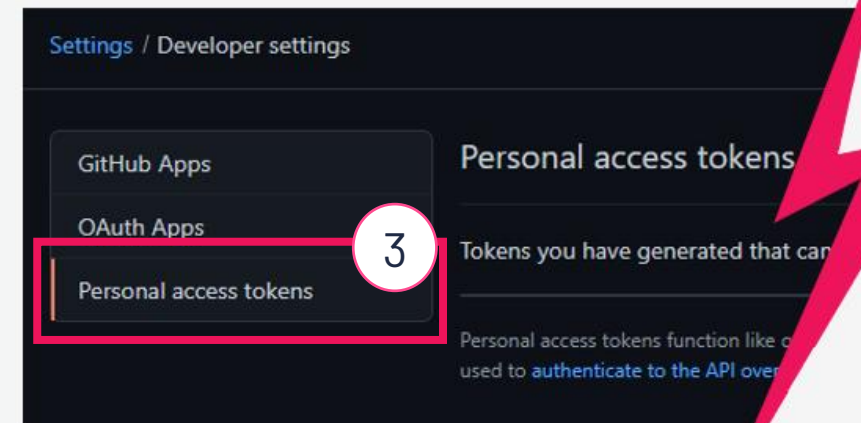
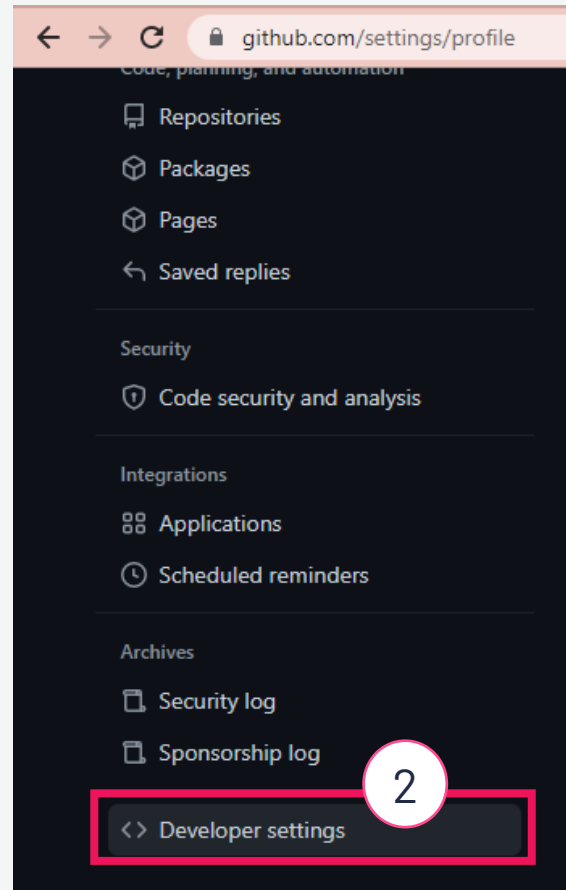
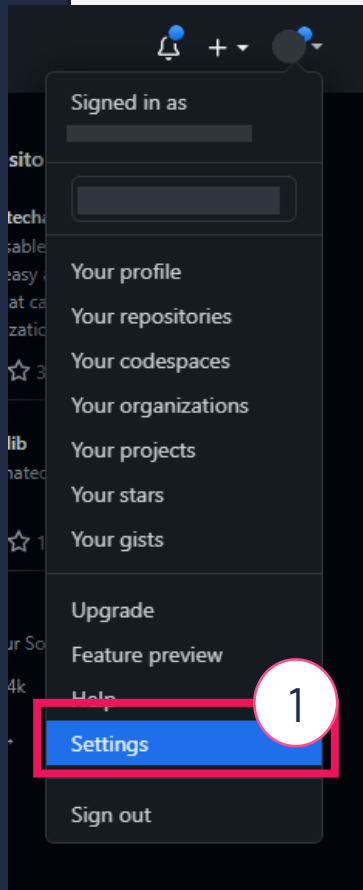
Como gerar seu token
de acesso



Training wheels - Antes

Gere o seu token de acesso. Acesse o seguinte caminho no GitHub:

Configurações (1) > Configurações de desenvolvedor (2) > Token de acesso pessoal (3) > Gerar novo token (4)



ANTES



Training wheels - Antes

Escolha um **nome** e as **permissões** do token (1)
Depois, clique em **"Gerar token"** (2)

Settings / Developer settings

GitHub Apps
OAuth Apps
Personal access tokens

New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

maquina-da-faculdade

What's this token for?

☒ Full control of the enterprise

☒ manage_runners:enterprise Manage enterprise runners and runner-groups

☒ manage_billing:enterprise Read and write enterprise billing data

☒ read:enterprise Read enterprise profile data

☒ admin:gpg_key Full control of public user GPG keys (Developer Preview)

☒ write:gpg_key Write public user GPG keys

☒ read:gpg_key Read public user GPG keys

2 Generate token Cancel

ANTES



Training wheels - Antes

Copie o token gerado pelo GitHub e **cole em um local de fácil acesso** pois iremos usá-lo nos próximos passos.

Pode colar em um bloco de notas ou enviar por e-mail para você mesmo (e apague o arquivo do bloco de notas ou e-mail após o uso do token)

(Atenção: este token só será mostrado a você esta única vez) 

ANTES

Settings / Developer settings

GitHub Apps

OAuth Apps

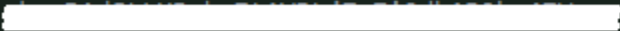

Personal access tokens

Personal access tokens

Generate new token Revoke all

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!

✓  

Delete

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

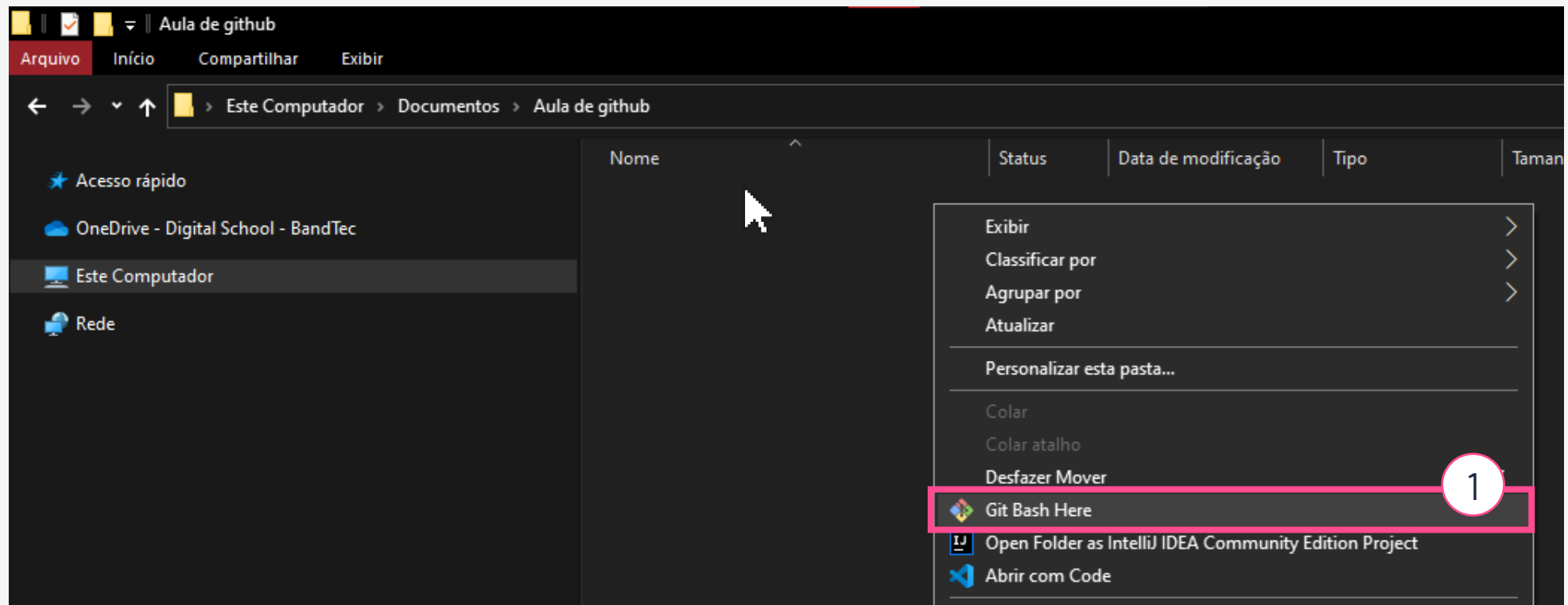
Antes dos comandos Git:

Como acessar o diretório/pasta
onde colocará seu projeto

Training wheels - Antes

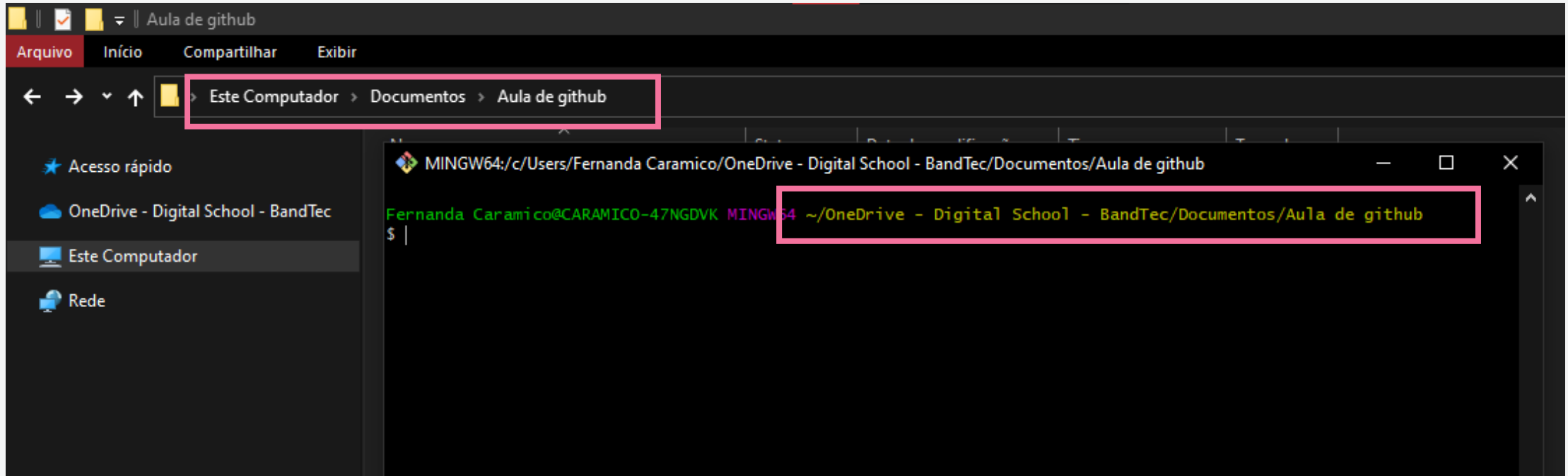
Escolha um diretório (pasta) na sua máquina onde você gostaria que seu projeto fosse localizado. Neste exemplo escolhi a pasta "Aula de github" que criei na pasta Documentos.

Para abrir o GitBash diretamente neste local, clique com o botão direito com mouse e escolha "**Git Bash Here**", como abaixo (1):



Training wheels - Antes

A janela do Git Bash deve ter sido exibida como abaixo, com o caminho em amarelo (como em destaque):

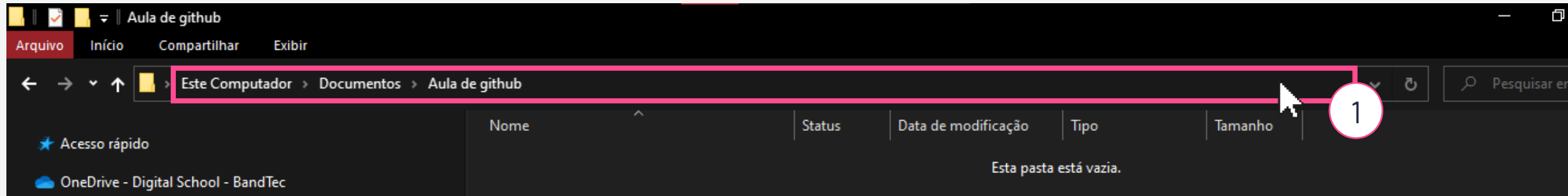


- ✓ Caso o atalho "Git Bash Here" do slide anterior não tenha aparecido para você, prossiga para o próximo passo.
- ✓ Caso tenha aparecido e você tenha conseguido acessar a janela do Git Bash como a imagem acima, vá até o passo "Durante - Os comando Git"

Training wheels - Antes

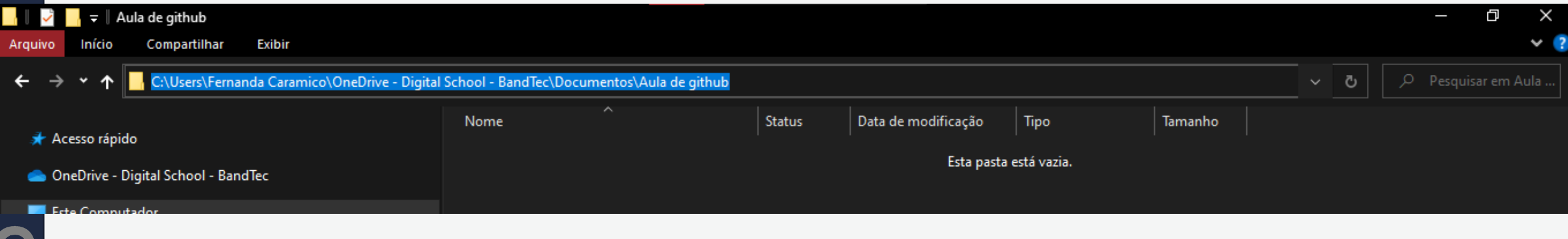
Escolha um diretório (pasta) na sua máquina onde você gostaria que seu projeto fosse localizado. Neste exemplo escolhi a pasta "Aula de github" que criei na pasta Documentos.

Acesse a pasta e clique com o botão esquerdo do mouse no caminho desta pasta (1):

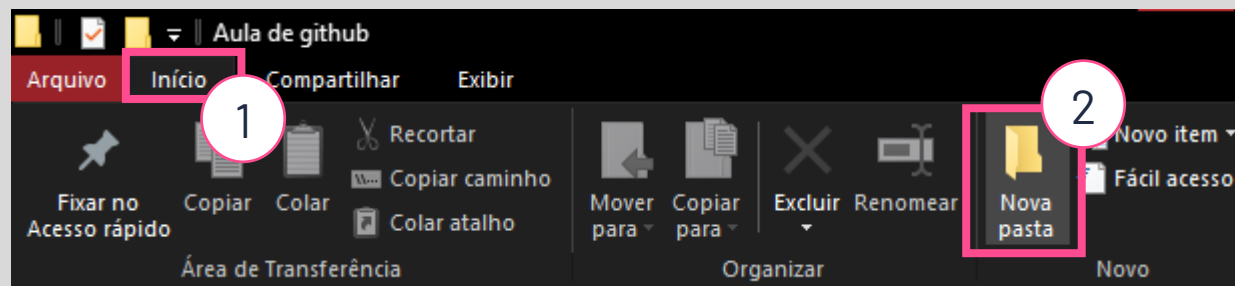


Training wheels - Antes

O caminho deve ser parecido com este abaixo e ficar azul. Este é o caminho do seu diretório escolhido na sua máquina. Copie com CTRL+C ou com o botão direito do mouse em "copiar". Precisaremos deste caminho no GitBash.



Se o caminho ao clicar for apenas "Documentos" ou "Downloads", não será possível seguir este passo a passo. Crie então uma pasta dentro de "Documentos" ou "Downloads" clicando no ícone como abaixo (1) e (2), acesse esta nova pasta e tente novamente.



Training wheels - Antes

Abra o GitBash e execute o comando abaixo:

cd oCaminhoQueVoceCopiouAntes

O comando **cd** significa
"**change directory**", ou
"mudar de diretório" (pasta)



Você pode colar o caminho copiado clicando com o botão direito do mouse no terminal e então escolhendo "**Paste**" (**Colar**) ou pelo **atalho Shift+Insert**:

Esta linha no seu computador pode (e deve) estar diferente!



```
MINGW64:/c/Users/Fernanda Caramico
Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~
$ cd
```

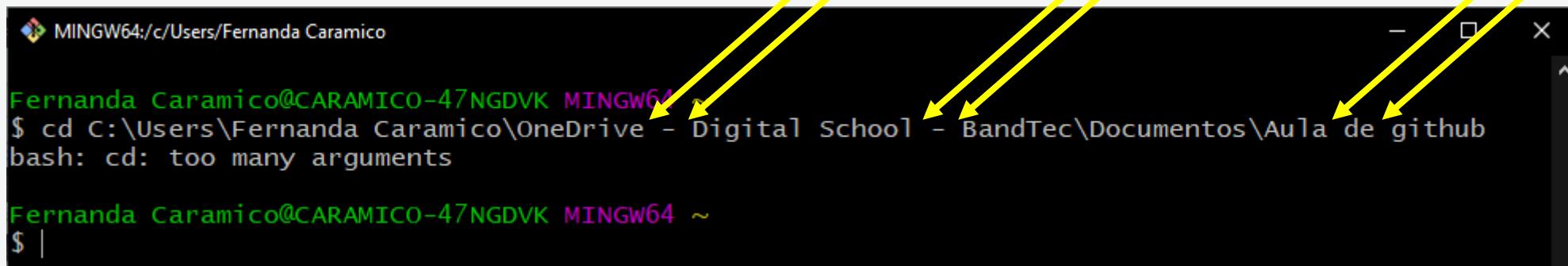
Open	
Copy	Ctrl+Ins
Paste	Shift+Ins
Select All	

```
MINGW64:/c/Users/Fernanda Caramico
Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~
$ cd C:\Users\Fernanda Caramico\OneDrive - Digital School - BandTec\Documentos\Aula de github|
```

Tecle [Enter]

Training wheels - Antes

Se você receber o erro *"too many arguments"* é porque há espaços no seu caminho.



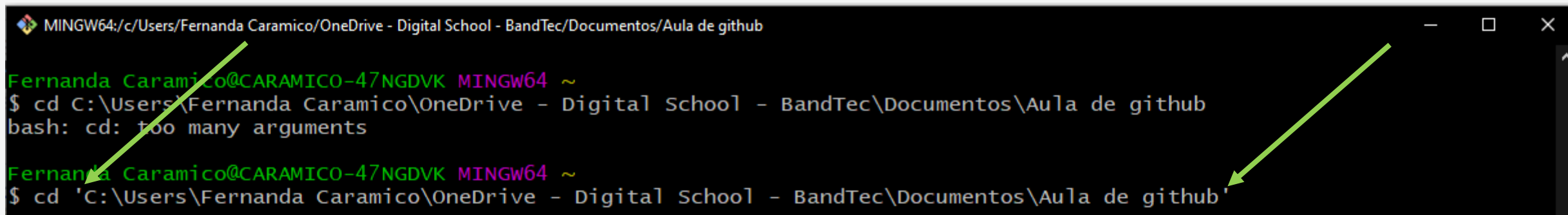
```
MINGW64:/c/Users/Fernanda Caramico

Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~
$ cd C:\Users\Fernanda Caramico\OneDrive - Digital School - BandTec\Documentos\Aula de github
bash: cd: too many arguments

Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~
$ |
```

Four yellow arrows point to the spaces in the file path: "Digital School", "BandTec", "Documentos", and "Aula de github".

Tente novamente mas, agora, coloque o caminho entre aspas simples.



```
MINGW64:/c/Users/Fernanda Caramico/OneDrive - Digital School - BandTec/Documentos/Aula de github

Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~
$ cd C:\Users\Fernanda Caramico\OneDrive - Digital School - BandTec\Documentos\Aula de github
bash: cd: too many arguments

Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~
$ cd 'C:\Users\Fernanda Caramico\OneDrive - Digital School - BandTec\Documentos\Aula de github'
```

A green arrow points to the single quotes around the entire file path in the command.

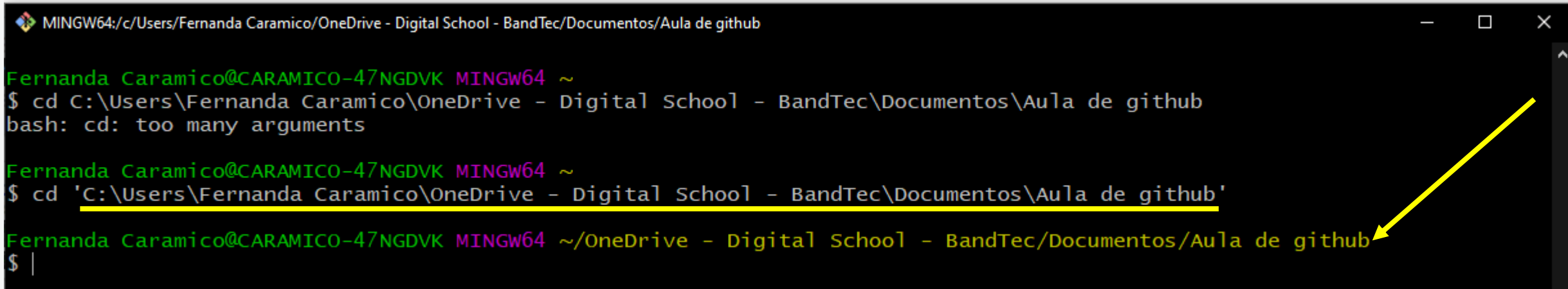
Pratique navegar pelo terminal usando o teclado, evite o mouse.

Você pode repetir os comandos anteriores clicando na seta para cima do seu teclado, assim não precisa digitar tudo de novo.



Training wheels - Antes

Quando o caminho copiado aparecer acima da linha que permite digitar (aqui, em amarelo), você conseguiu acessar o diretório pelo GitBash!



```
MINGW64:/c/Users/Fernanda Caramico/OneDrive - Digital School - BandTec/Documentos/Aula de github

Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~
$ cd C:\Users\Fernanda Caramico\OneDrive - Digital School - BandTec\Documentos\Aula de github
bash: cd: too many arguments

Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~
$ cd 'C:\Users\Fernanda Caramico\OneDrive - Digital School - BandTec\Documentos\Aula de github'

Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~/OneDrive - Digital School - BandTec/Documentos/Aula de github
$ |
```

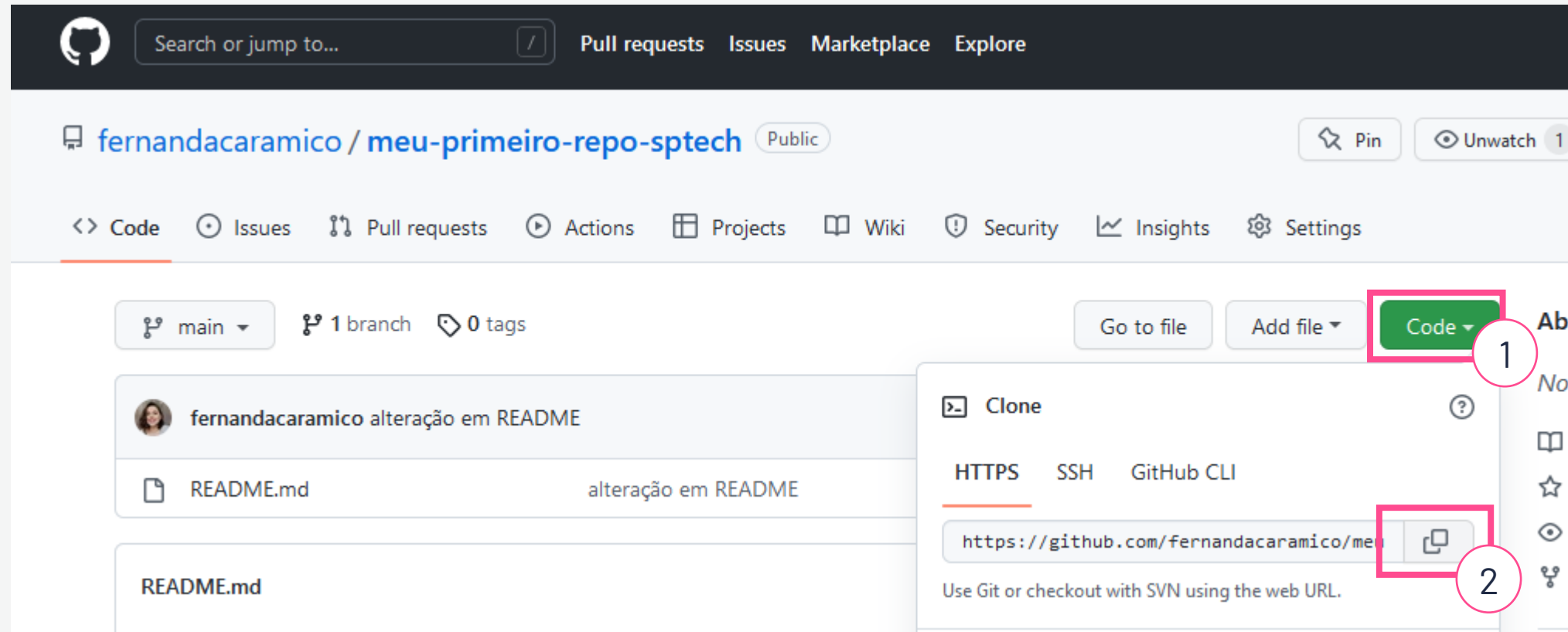
Durante

Os comandos Git


Training wheels

Feito o passo a passo em "Antes", continue:

Acesse o GitHub e copie a URL HTTPS do seu diretório:



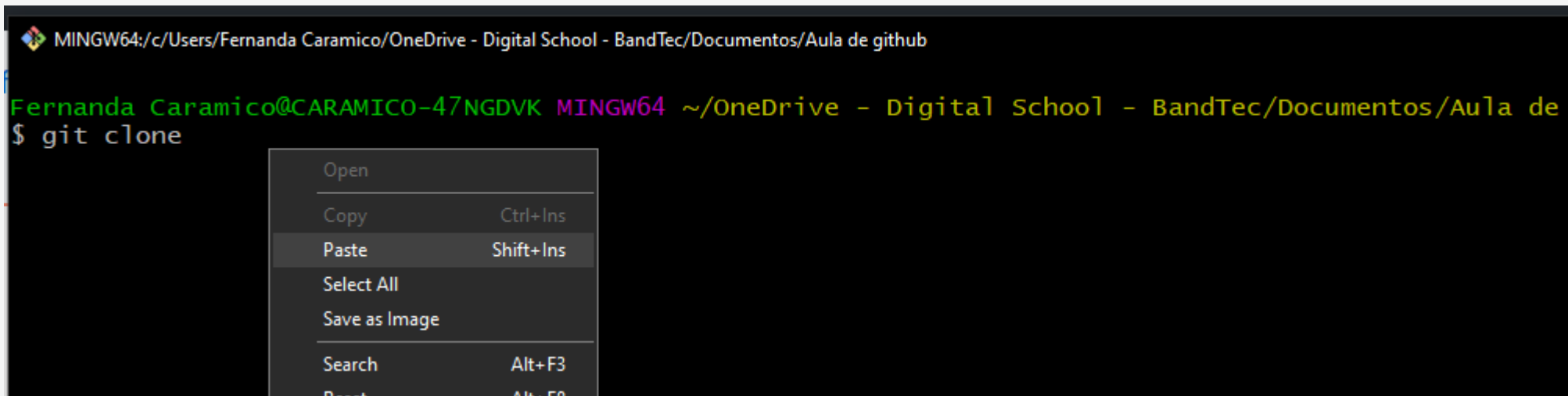
The screenshot shows the GitHub interface for a repository named 'meu-primeiro-repo-sptech' by user 'fernandacaramico'. The 'Code' button is highlighted with a red box and a circled '1'. A dropdown menu is open, showing the 'Clone' section with the 'HTTPS' option selected. The URL 'https://github.com/fernandacaramico/meu-primeiro-repo-sptech' is displayed, and a copy icon (two overlapping squares) is highlighted with a red box and a circled '2'. The repository content shows a 'README.md' file with a recent commit 'alteração em README'.

Este ícone  já faz a cópia do link para sua Área de Transferência, seu CTRL+C!

Training wheels

Usando o terminal de comando (GitBash, se Windows e Terminal se Linux ou MacOS), traga o repositório remoto para o local que acessamos nos passos anteriores.

Digite **git clone** e cole ao lado a URL copiada:



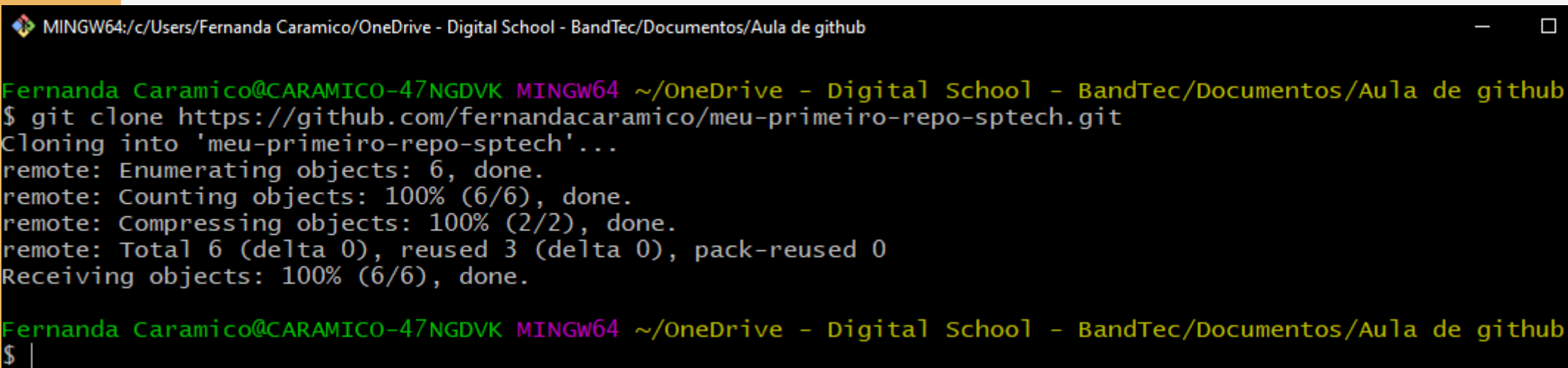
```
MINGW64:/c/Users/Fernanda Caramico/OneDrive - Digital School - BandTec/Documentos/Aula de github
Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~/OneDrive - Digital School - BandTec/Documentos/Aula de
$ git clone
```

The screenshot shows a terminal window with a right-click context menu open over the command line. The menu options are: Open, Copy (Ctrl+Ins), Paste (Shift+Ins), Select All, Save as Image, Search (Alt+F3), and Reset (Alt+F8). The 'Paste' option is highlighted.

Lembrando: você pode colar a URL copiada clicando com o botão direito do mouse no terminal e então escolhendo **"Paste" (Colar)** ou pelo **atalho Shift+Insert**.

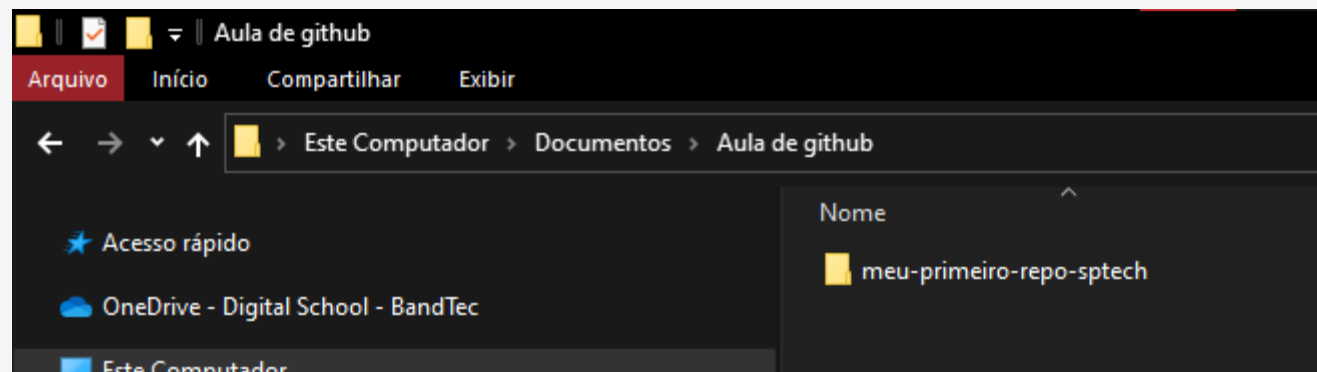
Training wheels

Seu clone terá executado com sucesso quando a tela ficar como a seguir:



```
MINGW64:/c/Users/Fernanda Caramico/OneDrive - Digital School - BandTec/Documentos/Aula de github
Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~/OneDrive - Digital School - BandTec/Documentos/Aula de github
$ git clone https://github.com/fernandacaramico/meu-primeiro-repo-sptech.git
Cloning into 'meu-primeiro-repo-sptech'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 6 (delta 0), reused 3 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), done.
Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~/OneDrive - Digital School - BandTec/Documentos/Aula de github
$ |
```

E seu diretório recém
clonado irá aparecer
no explorador de
Arquivos!



DURANTE

O comando **ls** significa “**list**”, ou “listar” e traz o conteúdo exibido no caminho em que você está no momento.



O comando **cd** significa “**change directory**”, ou “mudar de diretório” (pasta)

Training wheels

Vamos acessar este diretório recém clonado pelo GitBash, para conseguirmos efetuar os comandos git para adicionar, “comitar” e enviar nossas alterações ao GitHub (o repositório remoto). Primeiro, vamos ver o que há no caminho em que estamos, usando o comando ls:

```
Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~/OneDrive - Digital School - BandTec/Documentos/Aula de g
$ ls
meu-primeiro-repo-sptech/
```

Com **ls** você lista os arquivos e diretórios que estão no caminho atual, ou seja, o que está em amarelo. Ao executar o comando...

```
Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~/OneDrive - Digital School - BandTec/Documentos/Aula de g
$ ls
meu-primeiro-repo-sptech/
```

... neste diretório, há apenas o repositório que trouxemos do GitHub! Vamos “entrar” nele usando o **cd**:

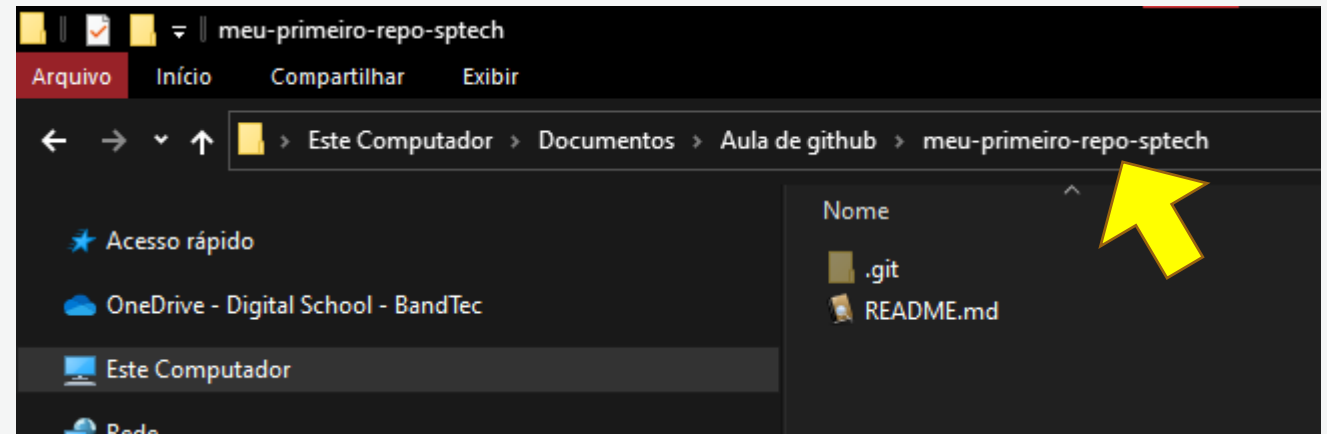
```
Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~/OneDrive - Digital School - BandTec/Documentos/Aula de g
$ cd meu-primeiro-repo-sptech/
```

Training wheels

Nosso caminho agora mudou e inclusive contém a nossa **branch*** entre parênteses:

```
Fernanda Caramico@CARAMICO-47NGDVK MINGW64 ~/OneDrive - Digital School - BandTec/Documentos/Aula de github/meu-primeiro-repo-sptech (main)
$ |
```

Estar neste caminho (acima - GitBash) é como estar neste caminho (abaixo - Explorador de Arquivos):



* **branch**: galho, ramo. Veremos mais à frente, mas agora é importante saber que é a mesma palavra presente no GitHub

Podemos agora fazer as alterações desejadas no nosso código!

Training wheels

Quando estiver no diretório desejado, prossiga:

Execute os comandos:

git pull *para trazer alterações ao repositório local*
~~~ faça suas as alterações ~~~

git add . *para adicionar alterações em pacote para serem enviadas*

git commit -m 'mensagem' *para fechar o pacote de alterações*

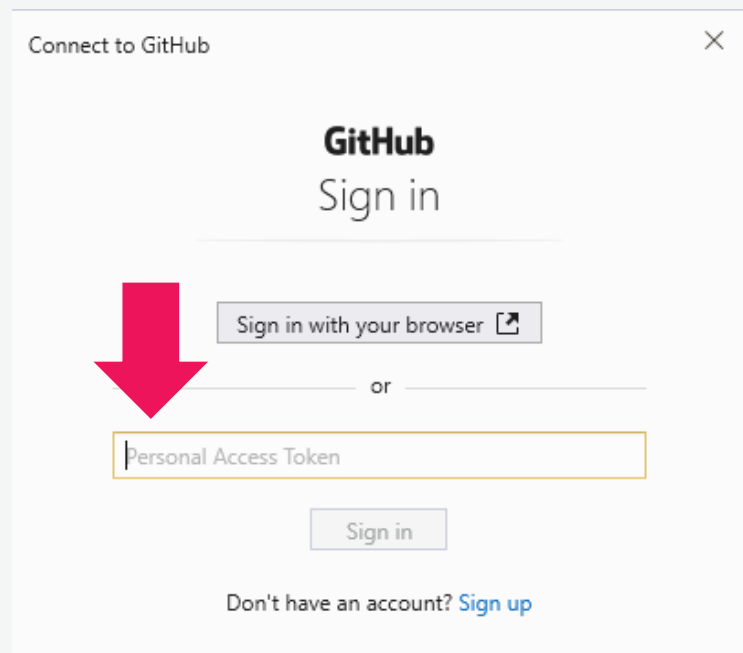
git push *para enviar as alterações feitas no repositório local ao repositório remoto.*

Utilize o Token gerado (próximo slide)

Training wheels

Como utilizar o Token gerado?

Ao efetuar um comando que precisa de autenticação como o git push, uma janela como as abaixo será aberta. Escolha a opção de Token como abaixo e preencha o campo com o Token gerado no GitHub e copiado por você.



Connect to GitHub

GitHub
Sign in

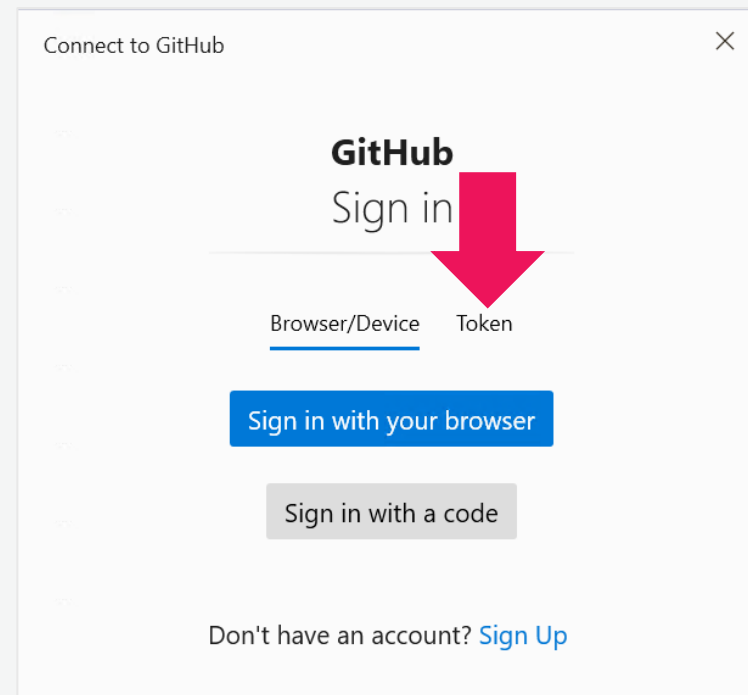
Sign in with your browser

or

Personal Access Token

Sign in

Don't have an account? [Sign up](#)



Connect to GitHub

GitHub
Sign in

Browser/Device Token

Sign in with your browser

Sign in with a code

Don't have an account? [Sign Up](#)

Depois dos comandos Git:

Como limpar suas credenciais

Training wheels

Remova suas credenciais de acesso

```
git config --global --unset user.name
```

```
git config --global --unset user.email
```

Verifique que suas alterações foram efetuadas com sucesso:

```
git config --global user.name
```

- *Aqui não deve ser exibido mais seu nome de usuário, mostrando nada.*

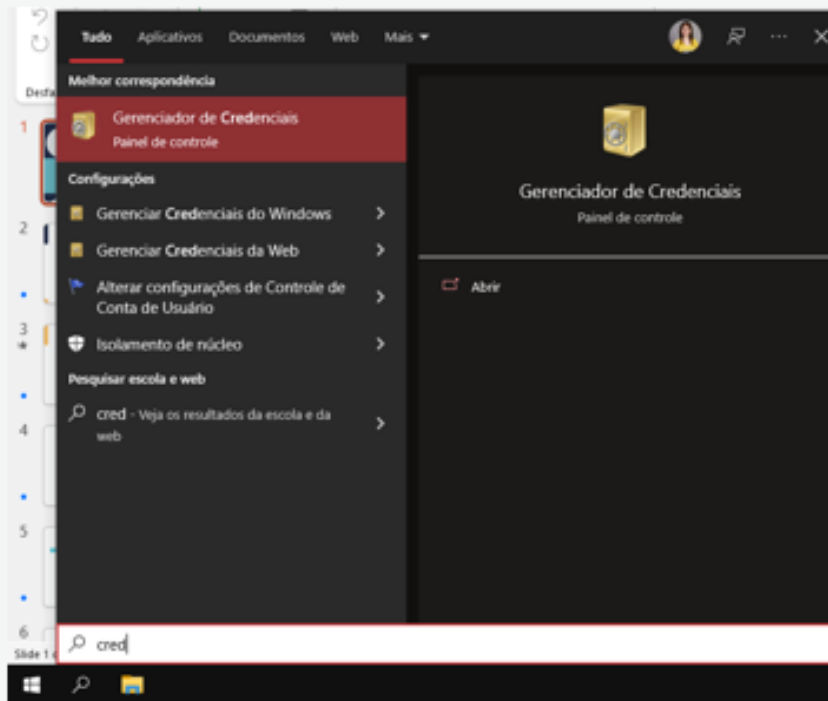
```
git config --global user.email
```

- *Aqui não deve ser exibido mais seu e-mail, mostrando nada.*

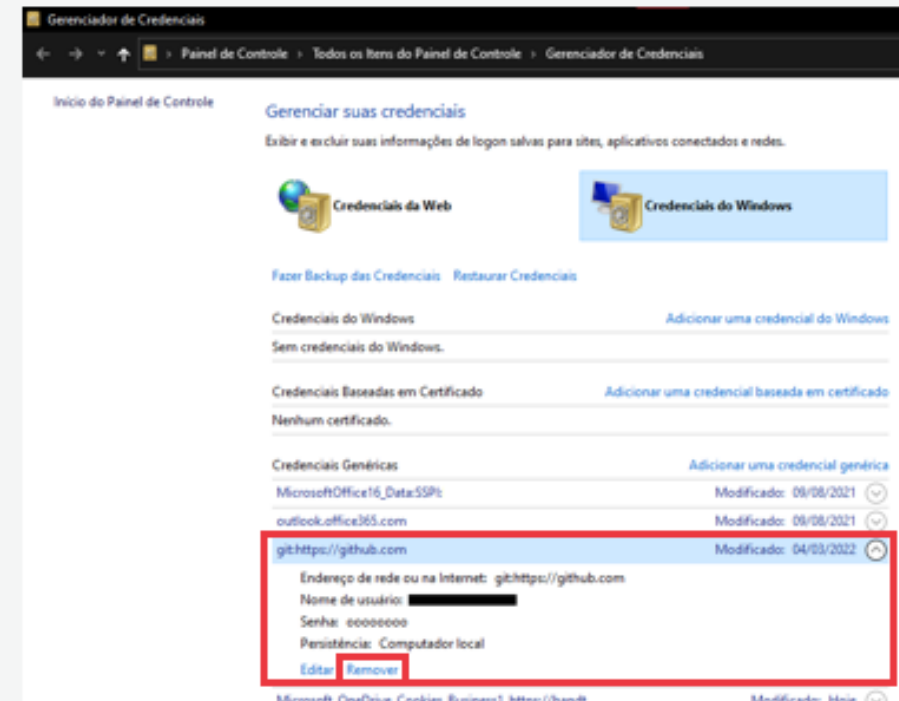
Training wheels

No Windows, verifique se há algum registro em Gerenciador de Credenciais e remova:

1. Busque por “Gerenciador de Credenciais”



2. Identifique a credencial “github” e clique em “Remover”



Depois dos comandos Git:

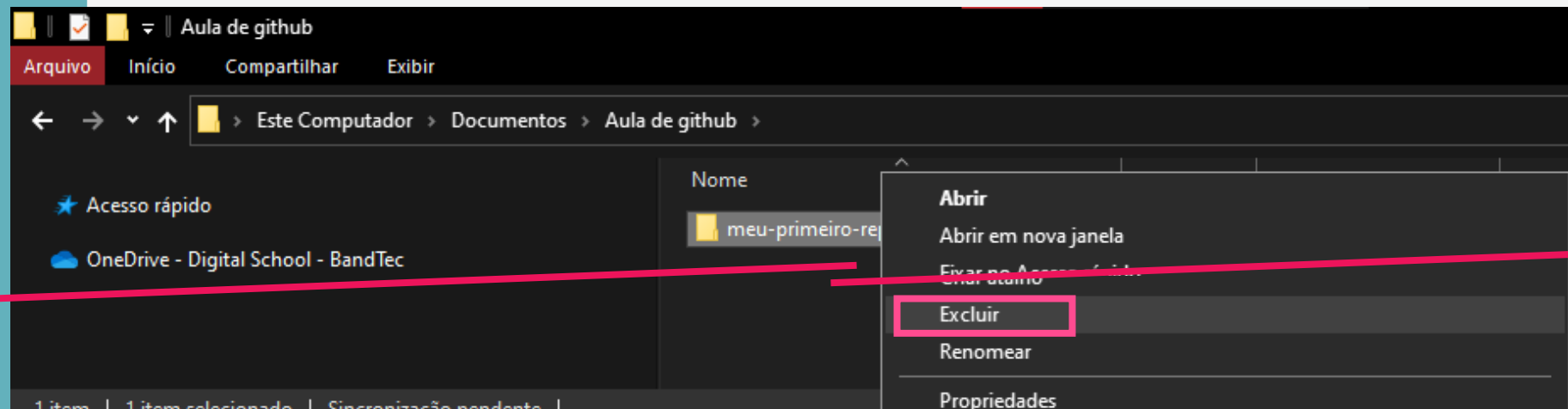
Como apagar o diretório/pasta
onde colocou seu projeto

Training wheels

Se você estiver utilizando uma das máquinas da faculdade, você deve apagar seus arquivos assim que não for mais utilizá-la.

Se você está utilizando sua máquina pessoal, você pode apagar os arquivos ao efetuar os comandos git pois o código desenvolvido já está salvo no servidor do GitHub.

Para apagar diretórios/pastas no Windows: selecione a pasta através do explorador de arquivos e clique com o botão direito em "Excluir", ou tecle em [delete]



Agradeço
a sua atenção!



SÃO
PAULO
TECH
SCHOOL