

Automação Cypress - Aula 0

criado: 14/12/2023

Objetivos

Nesta aula, aprenderemos os fundamentos iniciais da automação com Cypress. Abordaremos a instalação das ferramentas essenciais, como Visual Studio Code, Git e Node.js, que são cruciais para o desenvolvimento e execução de scripts de automação. Em seguida, exploraremos o processo de criação de um repositório no GitHub, fornecendo uma base sólida para o versionamento do código.

Ao longo do tutorial, guiaremos você na instalação das ferramentas, verificação das instalações e na criação do seu próprio repositório no GitHub. Também abordaremos o procedimento para baixar o repositório para sua máquina local usando comandos Git. Além disso, apresentaremos exercícios práticos para reforçar os conceitos aprendidos, como a criação e o download de múltiplos repositórios no GitHub. Para aqueles que desejam ir além, incluímos um exercício extra sobre a exclusão de repositórios no GitHub. Vamos começar!

Instalações

Ferramentas necessárias



Visual Studio Code (Editor de código):

<https://code.visualstudio.com/>



Git (Controlador de versão):

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



NodeJs (ambiente de execução JavaScript no servidor):

<https://nodejs.org/en>

Repositório

Ferramentas necessárias



Cadastrar-se em uma plataforma de hospedagem de código-fonte.

<https://github.com/>

Versões aplicadas em 14/12/2023

VScode: 1.85.1 | Git: 2.33.0.windows.2 | Node.js: v19.6.0

É hora de executar

Baixando e instalando as ferramentas

Baixe as ferramentas conforme links apresentados acima e instale da forma mais simples, apenas dando continuidade a instalação para todas as ferramentas.

Verificando instalações

Após a instalação das ferramentas vamos validar se estão instaladas corretamente utilizando qualquer prompt de comando, abaixo apresento códigos de verificação para cada uma delas.



Visual Studio Code:

Comando: `code --version`

```
C:\Users>code --version
1.85.1
0ee08df0cf4527e40edc9aa28f4b5bd38bbff2b2
x64
C:\Users>_
```

Resultado esperado: Versão da instalação e código hash (sopa de letras)



Git (Controlador de versão):

Comando: `git --version`

```
C:\Users>git --version
git version 2.33.0.windows.2
C:\Users>_
```

Resultado esperado: Versão da instalação



NodeJs (ambiente de execução JavaScript no servidor):

Comando: `node --version`

```
Prompt de Comando

C:\Users>node --version
v19.6.0

C:\Users>
```

Resultado esperado: Versão da instalação

Github

Criando meu repositório

Agora com acesso a ferramenta e com os devidos dados de acesso, vamos criar um repositório.

Abaixo apresento imagem de exemplo com configuração de abertura para o repositório.

Owner * / Repository name *
DioACCT / learn_cypress
learn_cypress is available.

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

Description (optional)
Projeto com o propósito de compartilhar ensinamentos sobre automação cypress do zero

☒ 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](#).

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

Create repository

Campos:

Repository name*:

Nome do seu repositório, campo obrigatório.

Description:

Campo de descrição do repositório opcional, voce poderá descrever do que se trata esse repositório.

Público ou privado:

O repositório pode ser exclusivo para um ou um grupo de usuários, sinalizando o Private, e público, aberto a todos terem acesso ao repositório.

Add README file:

Marque o check para adicionar um arquivo Readme, geralmente utilizamos para descrever melhor informações pertinentes ao projeto. Neste caso, seria para informar como instalar dependências, versão da ferramenta, entre outras informações.

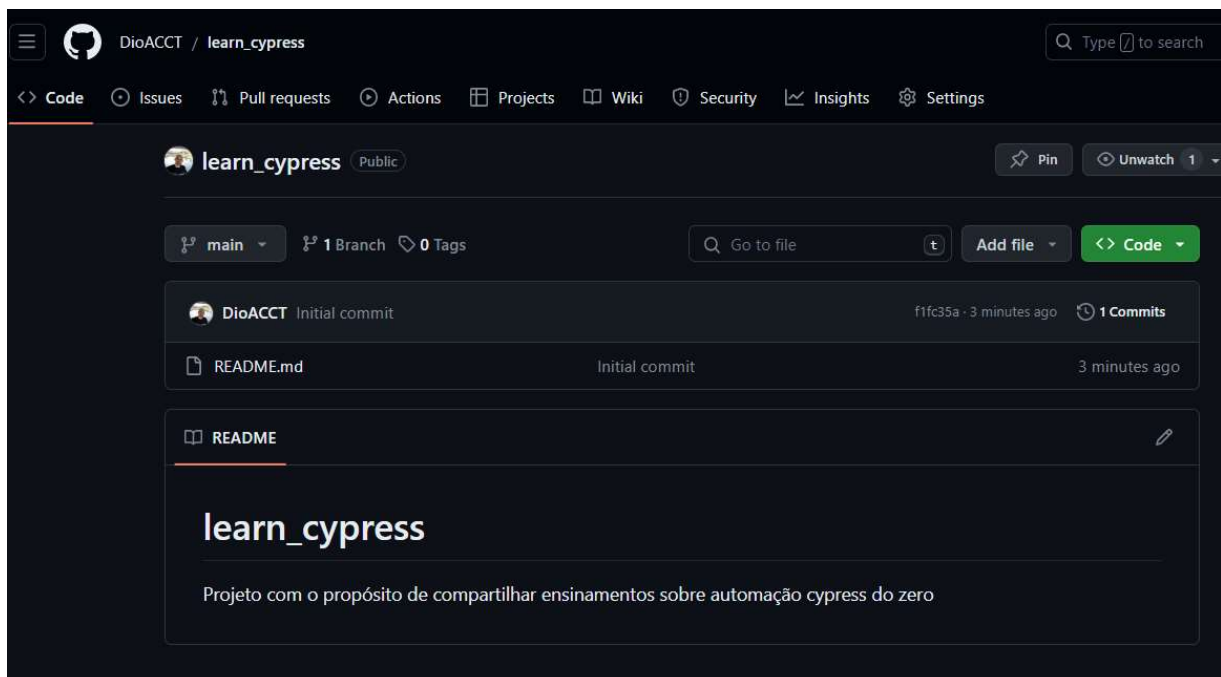
Add .gitignore:

Campo para informar arquivos que não deseje que sofram atualizações, nessa aula não utilizaremos, será melhor abordado em outro momento.

Choose a licence:

Aqui são disponibilizadas tipos descrição de licenças. No nosso caso, não precisaremos, pois nosso código é referente a automatização de processo.

Em seguida, clique em “[Create Repository](#)”



Baixando repositório para sua máquina

Uma vez que temos acesso ao repositório, precisamos baixa-lo para termos mais controle e poder e contribuir na atualização do projeto.

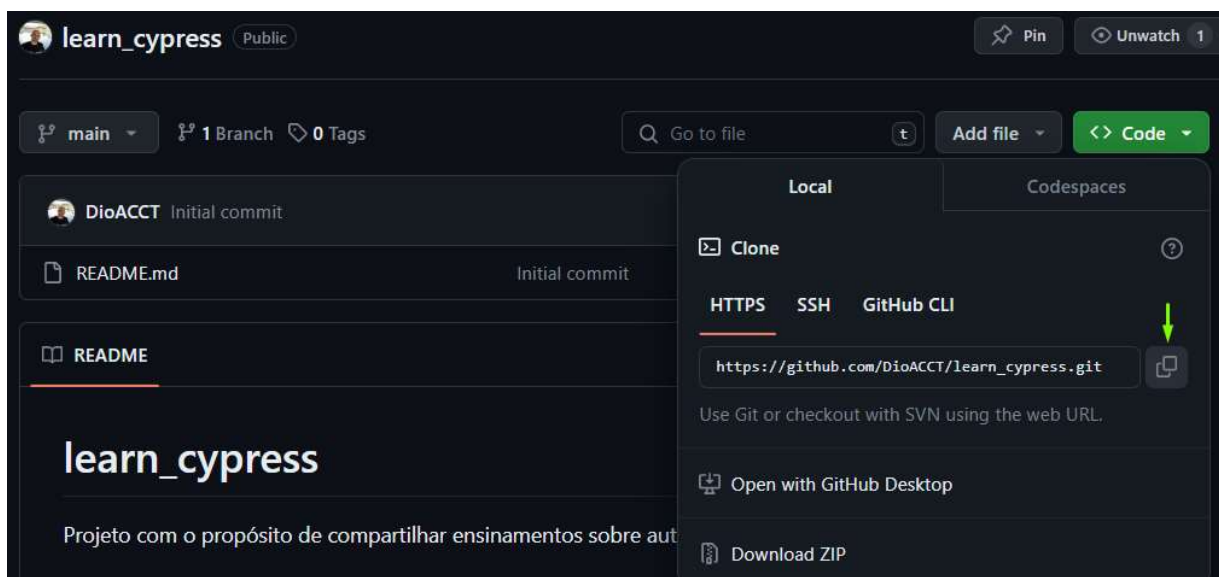
Esse processo que será apresentado, pode ser utilizado para quaisquer repositórios. Para aqueles com status “private” será necessário que tenha acesso aos mesmos.

Primeiros comandos Git

Clonando repositório vis HTTPS

No repositório git, clique no botão  e em HTTPS.

Em seguida copie o endereço fornecido, conforme apresenta a imagem abaixo.



No seu prompt de comando, na pasta desejada, vamos digitar nosso primeiro comando para baixar o repositório, conforme apresento o código abaixo.

git clone “comando copiado”



Informações extras

Primeiros Comandos prompt

Frequentemente vamos usar o prompt de comando para manipular nosso repositório, assim apresento os principais comandos para navegação entre pastas e subpastas, chamados de diretórios.

Listando pastas:

```
> dir
```

Acessando uma pasta:

```
> cd "nome da pasta"
```

Voltando uma pasta:

```
> cd ..
```

Exercícios de fixação

Crie três repositórios

No github exercite a criação de repositórios, todos com o arquivo README;

Baixe os três repositórios em uma única pasta

Aqui você irá salvar os repositórios em uma pasta de sua escolha.
Segue exemplo de comportamento esperado:

Pasta principal

- |_ Repositório 1
- |_ Repositório 2
- |_ Repositório 3

Exercício Extra ★

Descubra como excluir os repositórios criados no github.