

Estudo Autodidata Git e GitHub



Aula: 4

Tema: Instalações e configurações importantes

GitHub

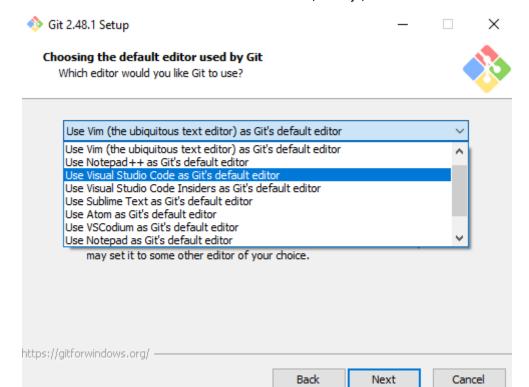
ESTUDOS GITHUB

Ferramentas a utilizar para instalação e configuração

- 1. Navegador
 - Google Chrome (Excelente para programação devido ao devTools)
 - Microsoft Edge (Navegador baseado no Chromium, software de codigo livre).
- 2. Editor de código
 - Visual Studio Code (Editor da Microsoft, compatível e integração com softwares que também são da Microsoft)
- 3. VCS
 - Git CSM (Editor de versionamento de codigo)
 - GitHub Desktop

Configurando o Git CSM

1. O editor é baseado na versão do Vs code, ou seja, "default" ou "insider"





ESTUDOS GITHUB

2. Caminho

Git 2.48.1 Setup

Adjusting your PATH environment

How would you like to use Git from the command line?



Use Git from Git Bash only

This is the most cautious choice as your PATH will not be modified at all. You will only be able to use the Git command line tools from Git Bash.

Git from the command line and also from 3rd-party software

(Recommended) This option adds only some minimal Git wrappers to your PATH to avoid cluttering your environment with optional Unix tools. You will be able to use Git from Git Bash, the Command Prompt and the Windows PowerShell as well as any third-party software looking for Git in PATH.

O Use Git and optional Unix tools from the Command Prompt

Both Git and the optional Unix tools will be added to your PATH. Warning: This will override Windows tools like "find" and "sort". Only use this option if you understand the implications.

https://gitforwindows.org/

Back

Next

Cancel

3. Backend de HTTPS



Choosing the SSH executable

Which Secure Shell client program would you like Git to use?



Use bundled OpenSSH

This uses ssh.exe that comes with Git.

Use external OpenSSH

NEW! This uses an external ssh.exe. Git will not install its own OpenSSH (and related) binaries but use them as found on the PATH.

https://gitforwindows.org/

Back

Next

Cancel



ESTUDOS GITHUB

4. Tipificação de linha



Configuring the line ending conversions

How should Git treat line endings in text files?



Checkout Windows-style, commit Unix-style line endings

Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrlf" is set to "true").

Checkout as-is, commit Unix-style line endings

Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Unix ("core.autocrlf" is set to "input").

Checkout as-is, commit as-is

Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform projects ("core.autocrlf" is set to "false").

https://gitforwindows.org/ -

Back	Next	Cancel

5. Terminal a utilizar (o mesmo Git instala)



Configuring the terminal emulator to use with Git Bash

Which terminal emulator do you want to use with your Git Bash?



Use MinTTY (the default terminal of MSYS2)

Git Bash will use MinTTY as terminal emulator, which sports a resizable window, non-rectangular selections and a Unicode font. Windows console programs (such as interactive Python) must be launched via `winpty` to work in MinTTY.

O Use Windows' default console window

Git will use the default console window of Windows ("cmd.exe"), which works well with Win32 console programs such as interactive Python or node.js, but has a very limited default scroll-back, needs to be configured to use a Unicode font in order to display non-ASCII characters correctly, and prior to Windows 10 its window was not freely resizable and it only allowed rectangular text selections.

https://gitforwindows.org/

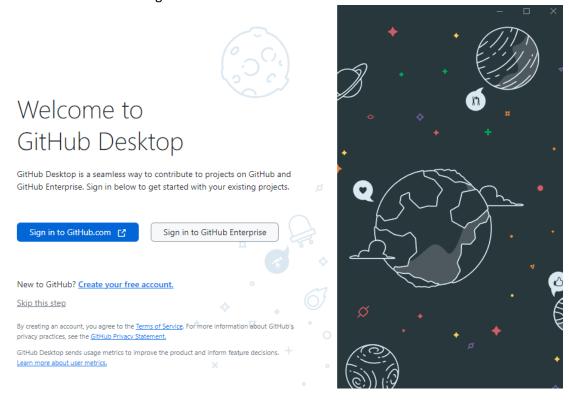
Back Next

Cancel



GitHub desktop

1. Entrar com a conta do git normalmente



2. Pronto

