

Controle de versão para iniciantes

Equipe E2PC - Unicentro



▶ ÍNDICE



01 Introdução

02

04

05

06

- Criando uma conta no Github
- **03** Autenticação via token
 - Configurando usuário no computador
 - Criando um repositório no Github
 - Usando o novo repositório criado
- 07 Comandos git



► INTRODUÇÃO

Git e Github

• O *Git* e o *GitHub* são duas ferramentas relacionadas, mas distintas, que desempenham papéis importantes no desenvolvimento de *software* colaborativo e no controle de versões.

INTRODUÇÃO

Git

- Git é um **sistema** de controle de versão distribuído projetado para projetos de qualquer tamanho
- Permite controlar e acompanhar alterações em arquivos e diretórios ao longo do tempo
- Facilita trabalho em equipe, colaboração e gerenciamento eficiente de código-fonte
- Possibilita rastreamento de alterações, criação de ramificações e mesclagem de versões do código.

INTRODUÇÃO

Github

- GitHub é uma plataforma de hospedagem de código-fonte baseada em nuvem.
- Permite que desenvolvedores e equipes compartilhem, colaborem e trabalhem em projetos de software usando Git.
- Fornece um repositório centralizado para projetos Git.
- Possibilita armazenar, gerenciar e compartilhar código com outros desenvolvedores





Criando uma conta no Github

- 1. Abra o seu navegador e vá para o site do GitHub: https://github.com
- 2. Na página inicial do GitHub, você verá um campo de inscrição no canto superior direito.



Criando uma conta no Github

- 3. Preencha as informações necessárias:
 - Escolha um nome de usuário: Digite um nome de usuário único que você deseja usar no GitHub.
 - Digite seu endereço de e-mail: Insira um endereço de e-mail válido que você tenha acesso.
 - Crie uma senha: Escolha uma senha segura para proteger sua conta.







Desde 2021 o GitHub tornou obrigatório o uso do token para autenticar operações. Para usar o token é necessário estar com seu e-mail verificado na plataforma. Após isso siga os seguintes passos como cita a documentação oficial:

- No canto superior direito clique na foto do seu perfil e procure configurações.
- Após isso vá até o final da página de configurações e na barra lateral esquerda, clique em Configurações do desenvolvedor.

	yorranan Yorranan	
	🚔 Learning	-
	A Your profile	
	 ☐ Your repositories [1] Your projects ⊟ Your codespaces ☆ Your stars ♡ Your sponsors 	
	☑ Your gists	
	Upgrade Try Enterprise Try Copilot	
	A Feature preview	_
	题 Settings	
roblemas-URI-Beecro epositório de problemas (GitHub Docs	
	유 GitHub Support	- /

3) Na barra lateral esquerda, em Personal access tokens, clique em Tokens clássicos.

E Settings / Developer Settings		Q Type [] to search	<u>}</u> + • C) n e 🍥
日 部 GitHub Apps	GitHub Apps	ands GitHub? Register a new GitHub App to get st	New GitHub App	
Personal access tokens Fine-grained tokens Tokens (dassic)	the GitHub API. You can also read more about buildin	g GitHub Apps in our developer documentation.		
🜍 © 2023 GitHub, Inc.	Terms Privacy Security Status Docs Contact Gi	Hub Pricing API Training Blog About		

4) Clique em Gerar novo Token

E C Settings / Developer Settings		Q Type 🛛 to search	> + • 0 n @ ()
器 GitHub Apps	Personal access tokens (classic)		Generate new token - Revoke all
 Auth Apps Personal access tokens 	Tokens you have generated that can be used to ac	cess the GitHub API.	
Fine-grained tokens (Beta) Tokens (classic)	Windows git — admin:enterprise, admin:gpg_key, admin: admin:repo_hook, admin:ssh_signing_key, audit_log, codes,		Last used within the last week Delete

5) Aparecerá uma tela com diversas opções.

		Q Type () to search	<u>></u> + •) (n) (n) (n) (n)				
GitHub Apps	New personal access	s token (classic)					
OAuth Apps Personal access tokens Fine-grained tokens Tokens (dassic)	Personal access tokens (classic) fur over HTTPS, or can be used to auth Note	Personal access tokens (classic) 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.					
	What's this token for?						
	Expiration * 30 days						
	Select scopes Scopes define the access for perso	nal tokens. Read more about OAuth scopes.					
	🔲 геро						
	repo:status	Access commit status					
	repo_deployment						
	public_repo						
	🔲 repo:invite						
	security_events	Read and write security events					

6) Em Nota do token, insira uma nota para ele.

7) Em Expiração, selecione uma validação para o token.

8) Para ter acesso completo as funcionalidades marque todas as opções disponíveis, contudo se deseja personalizar adequadamente confira a documentação

		Q Type (7) to search			01	1 🗛 🧕
B GitHub Apps	New personal acces	ss token (classic)				
A OAuth Apps Personal access tokens ^ Fine-grained tokens Beta Tokens (classic)	Personal access tokens (classic) fr over HTTPS, or can be used to au Note	unction like ordinary OAuth access tokens. They can be us thenticate to the API over Basic Authentication.	sed instead of a passw	ord for Gi		
	What's this token for? Expiration * 30 days The token wi					
	Select scopes Scopes define the access for pers	onal tokens. Read more about OAuth scopes.				
	repo repo:status	Full control of private repositories Access commit status				
	 repo_deployment public_repo repo:invite 	Access deployment status Access public repositories Access repository invitations				
	security_events	Read and write security events				

9) Por fim, copie o token gerado, que será usado para autenticar as operações.

Some of the scopes you've selected are included in other scopes. Only the minimum set of necessary scopes has been saved.	E C Settings / Developer Settings		o search	+ • O n	
% GitHub Apps % OAuth Apps Personal access tokens Personal access tokens Fine-grained tokens Beta Tokens (classic)	Some of the scopes you've selected are included	in other scopes. Only the minimum set of necessary scopes has bee	n saved.		×
A OAuth Apps Personal access tokens Tokens you have generated that can be used to access the GitHub API. Fine-grained tokens Beta Tokens (classic) Make sure to copy your personal access token now. You won't be able to see it again!	器 GitHub Apps	Personal access tokens (classic)	Generate new token 🕶	Revoke all	
Fine-grained tokens Beta Tokens (classic) Make sure to copy your personal access token now. You won't be able to see it again!	 OAuth Apps Personal access tokens 	Tokens you have generated that can be used to access the GitHub	API.		
	Fine-grained tokens Beta Tokens (classic)	Make sure to copy your personal access token now. You won't	be able to see it again!		
✓ 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 Delete		✓ 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 ᢕ		Delete	





Configurando usuário no computador

• Através do PowerShell ou terminal é necessário realizar as configurações de uso, para isso é necessário usar os seguintes comandos:

git config --global user.name "Fulano de Tal"

git config --global user.email <u>fulanodetal@exemplo.br</u>

• Esses comandos inserem um nome de usuário e email para identificar quem modifica os códigos de maneira global.

Configurando usuário no computador

- É importante salientar que isso é a configuração do **Git** e não do GitHub
- Outro ponto importante é que esse comando deve ser executado toda vez que o usuário for usar o git, uma vez que se o usuário não estiver logado não poderá fazer alterações no repositório.
- Para fazer log out usar o comando:

git config --global --unset-all

Configurando usuário no computador

• Para evitar a necessidade de informar o token toda vez que fizermos operações usando o GitHub, você pode salvar no seu computador essa informação usando o seguinte comando:

git config --global credential.helper cache

• Após isso ele salvará a próxima credencial inserida pelo usuário. Exemplo:

PowerShell	×	🧿 peripecia@WIN-Yorranan: ~	× + ~
peripecia@WIN-Yorra peripecia@WIN-Yorra peripecia@WIN-Yorra peripecia@WIN-Yorra	nan:~ nan:~ nan:~ nan:~	\$ git configglobal \$ git configglobal \$ git configglobal \$	user.name "yorranan" user.email yorranan.almeida@gmail.com credential.helper cache



• No canto superior direito da página (de seu perfil), clique no botão "+", ao lado da sua foto de perfil, e selecione "Novo repositório" no menu suspenso.



۲

• Se você estiver na página inicial procure o botão verde escrito "novo" ou "new" localizado na barra lateral esquerda.

Top Repositories	Join GitHu	b Global Can	npus!				Latest ch	nanges Irdav
	Prepare for a	career in tech b	y joining GitHu	o Global Ca	ampus. G	lobal	GitH	lub Actions – Actions Runner
Second to the Table	Campus will r	nelp you get the cess to industry	tools events le	arning res	dge you r	need by	Gen	eral availability
🌸 yorranan/Hasn-Table	growing stud	lent community.		unnig res	ources u		e 2 da Gro	/s ago uped version updates for
🚯 LorenzoPilati/Torre-de-Hanoi			Breaking into tech: Le internship edition with Tr	d up your code with ioQuest	Learning by teachin your community - C	ng for Cassidy	Dep	endabot public beta
yorranan/Problemas-URI-Beecrowd	no la gerbejo with faciations & events.	Follow your Expert	Helen Huang Science & Technology Tai	Drown & Podcasta	Withers Special Events		• 2 day	/s ago R codec for 2EA are now cingle-
yorranan/yorranan.github.io	Popular offers yo	to have not claimed	Citize mare (d	Correct y	your lecal Expert		use	r codes for ZFA are now single-
🐳 yorranan/Tesauro	Constant Departure	💽 💿 🚩 🕂 🔕 ¥	🕑 🧿 79-	 View proje View proje View proje 	ects at our gallery tudent Gallery			
yorranan/Banco_de_Dados_Graduacao	Virbal event ki	a)		Learn more Click en an	re about an event		Secu	rity risk and coverage pages are
	2enaery 22, 2011 Level up your code with TwilloQuest	February 1, 2022 Othisb Cerepus Experts applications are open . Juan Pablo Roses Contes	Antificial Intelligence 6 assignments Due by May 1, 2021, 12:00 PST	Which a C Visit Group Claimed a Sine provise	Campus TV episode In Education on Neticn a Shudent Pack offer ar offers	Web Meet	View	changelog →
Recent activity			Wesk Five: Functions Don by Aug 16, 2021, 14:00 PS1			It is being partnered b Github- the biggest op		
yorranan/UpBringAcademy Pesquisa de Ferramentas	Join Global	Campus						

- Na página "Criar um novo repositório", preencha as informações básicas do seu repositório.
- No campo "Nome do repositório", digite um nome descritivo para o seu projeto.
- Escreva uma breve descrição do repositório no campo "Descrição" (opcional).
- Escolha se o repositório será público (visível para todos) ou privado (visível apenas para você ou pessoas específicas, se você tiver uma conta PRO).
- Se desejar, selecione a opção "Inicializar este repositório com um arquivo README" para criar automaticamente um arquivo README.md inicial para o seu repositório. O arquivo README.md é usado para fornecer informações sobre o projeto.
- Você também pode adicionar um arquivo de licença, um arquivo .gitignore específico e outros arquivos adicionais, se desejar.

• Após preencher as informações, clique no botão "Criar repositório". Seu repositório será criado no GitHub.

Owner *	Repository name *
🛞 yorranan	/ nome-do-repositorio
	🥝 nome-do-repositorio is available.
Great repository n	ames are short and memorable. Need inspiration? How about scaling-palm-tree ?
Description (optio	
Para nomes de r	epositórios recomendamos não usar acentos latinos ou caracteres especiais.
• Public	on the internet can see this repository. You choose who can commit.
O A Private You choo	
Initialize this repo	sitory with:
Add a READM	E file
Add .gitignore	
	· None 👻
Choose a license	
License: MIT Licens	e 🔹

• Agora, você pode adicionar arquivos ao seu repositório, cloná-lo para o seu computador local, fazer *commits* e *push* das suas alterações, e muito mais. O GitHub oferece instruções detalhadas para começar a trabalhar com o seu novo repositório, incluindo comandos Git que você pode usar!



• No Windows 11 pode-se abrir a pasta onde se deseja inserir o novo repositório, clicando com o botão direito selecione a opção de abrir no terminal. Em grande parte das distribuições existem opções similares usando a interface gráfica.

s\UNICENTRO\monitori	ia2023 × +					
① Novo ~		🖻 🗊 🕇 Cla	ssificar 🗸 🔳	Visualizar 🗸 🔸 🔸		
\leftrightarrow \rightarrow \checkmark \uparrow	Yorranan – Pessoal	Documentos > UNICENTRO	> monitoria2023			
sinício	Nome		Status	Data de modificação	Тіро	Tamanho
👻 🥌 Yorranan – Pessc	🧰 Cap 2 - Intro	88 Exibir		25/06/2023 09:31	Microsoft Edge PDF	242 KB
> 🧮 Área de Traball	🖻 Tutorial_Git_e	↑↓ Classificar por		29/06/2023 23:43	Documento do Micr	15 KB
> 🖹 Documentos	Tutorial_Git_C	🗄 Agrupar por		29/06/2023 23:29	Documento do Micr	12 KB
> 🔀 Imagens	🖻 Tutorial_Git_C	汐 Desfazer Mover	Ctrl+Z	29/06/2023 23:28	Texto OpenDocument	5 KB
> 📀 Música	🖻 UNIVERSIDAI	• Novo		29/06/2023 23:27	Texto OpenDocument	5 КВ
> 🗾 Vídeos		B Propriedades	Alt+Enter			
		도 Abrir no Terminal				
Area de Traba		🖸 Mostrar mais opções				
Documentos						

• No Windows poderá aparecer uma tela de autenticação similar a essa após usar git clone.

git clone <link>



• Se essa tela não aparecer por quaisquer motivos, o terminal perguntará seu usuário e você deve inserir o mesmo e-mail cadastrado no GitHub. Após isso ele deve pedir a senha ou autenticação, nesse caso cole o token que você copiou

PowerShell	× + ~				0	×
PowerShell 7.3.5 PS C:\Users\yorra\O Cloning into 'tutor	neDrive\Documents\UNIC ial_teste'	ENTRO\monitoria2023> git clone https://gi	ithub.com/yorranan/tutor	ial_teste.git		
		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				

• Após isso, qualquer modificação que você realizar na pasta clonada permanecerá localmente até que você envie um commit. Dentro da pasta do repositório que você clonou, a partir do terminal deve-se verificar as modificações usando o comando git status.

	Ø	
PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste> git status On branch main		
No commits yet		
Untracked files: (use "git add <file>" to include in what will be committed) documento de texto.txt</file>		
nothing added to commit but untracked files present (use "git add" to track) PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste>		

• Pode-se adicionar os arquivos individualmente ou todos através do git add *.

```
PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste> git add *
PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste> git stat
git: 'stat' is not a git command. See 'git --help'.
The most similar commands are
        status
        stage
        statsh
PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste> git status
        On branch main
```

No commits yet

```
Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: documento de texto.txt
```

PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste>

Feito isso, os arquivos em verde estão prontos para serem "empacotados" usando o ۲ comando commit.

PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste> git commit -m "A mensagem" [main (root-commit) dfb22bc] A mensagem

¹ file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 documento de texto.txt

PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste>

• Feito as operações acima podemos finalmente realizar "empurrar" nosso commit para o repositório no GitHub usando push.

PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste> git commit -m "A mensagem"
[main (root-commit) dfb22bc] A mensagem
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 1006444 documento de texto.txt
PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste> git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 224 bytes | 224.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/yorranan/tutorial_teste.git
 * [new branch] main -> main
PS C:\Users\yorra\OneDrive\Documents\UNICENTRO\monitoria2023\tutorial_teste>

• Seu projeto agora está no Github!



• Os comandos do Git são um conjunto de instruções utilizadas para interagir com o sistema de controle de versão distribuído Git, todos os comandos começam com a palavra git, usando o comando git –help no terminal é possível ver mais comandos e suas finalidades , os mais usados são:

git init
Sintaxe: git init

Descrição: Inicializa um novo repositório Git no diretório atual.

Esse comando é usado para inicializar um novo repositório *Git* em um diretório local. Ele cria uma estrutura .git, que é responsável por armazenar o histórico de alterações e os metadados do projeto.

git clone

Sintaxe: git clone <URL do repositório>

Descrição: Clona um repositório existente do Git para o seu computador local. Isso cria uma cópia local do repositório completo, incluindo todo o histórico de alterações.

Exemplo: git clone https://github.com/usuario/repositorio.git

git status Sintaxe: git status

Descrição: Mostra o status atual do seu repositório *Git*, exibindo as alterações pendentes, os arquivos adicionados, entre outros detalhes

git add Sintaxe: git add <arquivo>

Descrição: Adiciona arquivos ao índice do *Git* para serem incluídos no próximo *commit*. O índice funciona como uma área de preparação para as alterações que você deseja commitar.

Exemplo: git add arquivo.txt

git rm
Sintaxe: git rm <arquivo>

Descrição: É usado para remover arquivos do diretório de trabalho e do índice do *Git*, ele pode ser usado para excluir arquivos que não são mais necessários no repositório.

Exemplo:git rm arquivo.txt

git commit

Sintaxe: git commit -m "Mensagem do commit"

Descrição: Grava um *snapshot* das alterações feitas nos arquivos adicionados ao índice. É uma forma de confirmar as alterações e torná-las parte do histórico do projeto. Você pode adicionar uma mensagem descritiva ao *commit* para registrar o propósito das alterações.

Exemplo: git commit -m "Adiciona novas funcionalidades"

git push

Sintaxe: git push <repositório-remoto> <branch>

Descrição: Envia as alterações locais para um repositório remoto. É usado para compartilhar as alterações feitas no seu projeto com outros colaboradores ou com o repositório central.

Exemplo: git push origin main

Links úteis

https://education.github.com/git-cheat-sheet-education.pdf

https://git-scm.com/doc

Links úteis

https://education.github.com/git-cheat-sheet-education.pdf

https://git-scm.com/doc



Obrigado!

Equipe E2PC - Unicentro

