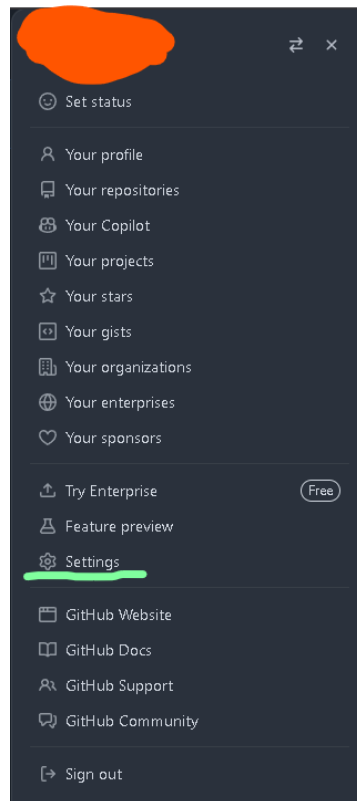


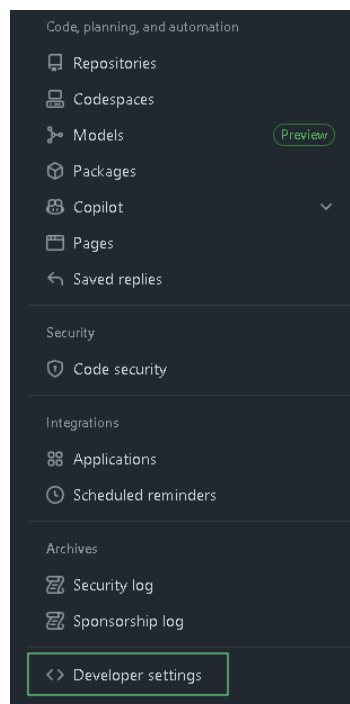
Guia para uso seguro do GIT nos laboratórios!

Para esta etapa é necessário que você já saiba iniciar um repositório GIT e entenda o funcionamento dos comandos básicos.

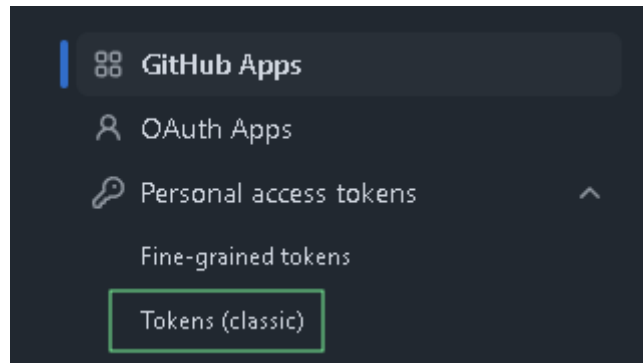
1. Ao acessar a pasta do seu repositório já com o terminal git bash aberto execute o seguinte comando:
 - a. **git config list**
2. Confira os seguintes parâmetros se estão vazios ou preenchidos:
 - a. **user.name**
 - b. **user.email**
3. Configure estes parâmetros para poder assinar os seus commits com:
 - a. **git config user.name "Nome de usuário"**
 - b. **git config user.email "Seu Email"**
 - i. **Utilize os mesmo que você cadastrou no github para que tudo funcione corretamente.**
4. Após esta configuração do repositório local, caso você queira commitar alterações no repositório da nuvem (GitHub) com git push e for a primeira vez que você faz isso, o github salvará suas credenciais no credentials manager da máquina. Após esse salvamento que ocorre de maneira automática, utilize o seguinte comando para apagar as suas credenciais do Windows:
 - a. **printf "host=github.com\nprotocol=https" | git-credential-manager erase**
5. Sem credenciais salvas no Windows, cada tentativa de push irá solicitar um usuário e token. Para gerar o token, logue na sua conta github, no canto superior direito clique na sua foto de perfil e selecione a opção **"Settings"**:



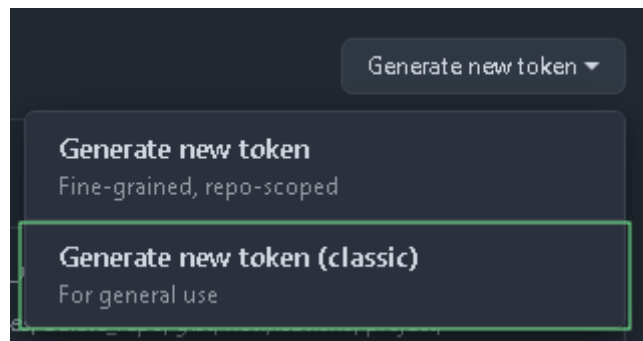
- a.
6. No menu esquerdo que irá aparecer selecione a opção “**developer settings**”



- a.
7. Após selecionada a opção, irá aparecer estas opções do menu superior do lado esquerdo:



- a.
8. Selecione a opção “Tokens (classic)”, e depois a opção “**Generate new tokens classic**”



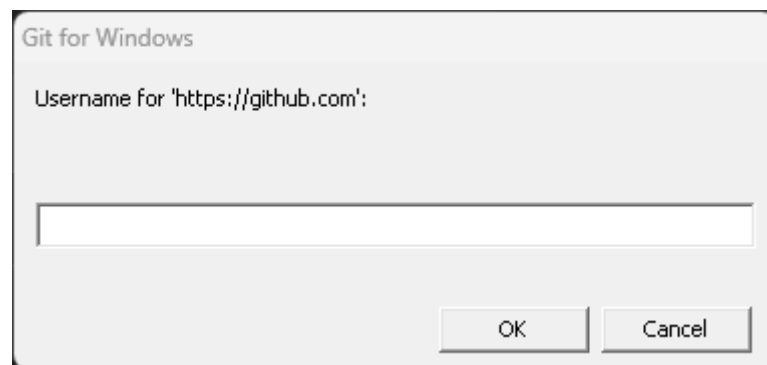
- a.
9. Após selecionar a opção “**Generate new token (classic)**” a seguinte tela irá surgir:

A dark-themed screenshot of the 'New personal access token (classic)' form. The title is 'New personal access token (classic)'. Below it is a paragraph explaining that these tokens function like ordinary OAuth access tokens. There's a 'Note' section with a text input field and the label 'What's this token for?'. The 'Expiration' section has a date picker set to '30 days (Sep 03, 2025)'. The 'Select scopes' section has a link 'Read more about OAuth scopes.' and a list of scopes with checkboxes: 'repo' (Full control of private repositories), 'repo:status' (Access commit status), 'repo_deployment' (Access deployment status), 'public_repo' (Access public repositories), 'repo:invite' (Access repository invitations), 'security_events' (Read and write security events), and 'workflow' (Update GitHub Action workflows).

- a.
- b. No campo “**Note**” digite um nome para o seu token.
- c. No Campo “**Expiration**” selecione uma data para seu token expirar (Não recomendo tokens sem data de expiração)

- d. No campo “Select scopes” Selecione todas as opções “**repo**” e a opção “**workflow**”, deixe todas as outras desmarcadas.
 - e. Role a tela até o final e selecione a opção “Generate Token”
10. Nesta etapa será apenas uma tela com seu token gerado. **PRESTE MUITA ATENÇÃO: SEU TOKEN SÓ SERÁ VISUALIZÁVEL NESTE MOMENTO**, por isso, copie-o e guarde como uma senha, em um pendrive, ou banco de senhas, ele será solicitado na etapa de **git push** durante o laboratório.
11. Durante o laboratório, após configurar seu repositório, no momento do push uma tela irá surgir pedindo seu nome de usuário (mesmo cadastrado no github) e sua senha. No local da senha você colará o token e a operação ocorrerá com sucesso.

a. Exemplo da tela:



12. Se suas credenciais não estiverem salvas no gerenciador de credenciais do windows, essa operação será solicitada a cada push que vocês executarem. Este é o melhor formato para os exercícios em laboratório com segurança, no seu computador pessoal, basta não executar o comando de apagar credenciais que elas continuaram salvas e não serão solicitadas a todo momento.
13. **Após o término da utilização do git em laboratório, recomendo a execução do comando descrito no passo 4 para garantir que não ficaram salvas credenciais na máquina.**
14. Uma vez realizado o passo a passo para configuração de um token, basta telo guardado para que você seja capaz de clonar repositórios em máquinas diferentes e fazer push das suas alterações para a nuvem com segurança apenas utilizando uma chave de acesso.