

08

Instalando o Git

09min

09

Para saber mais: git init

10

Faça como eu fiz: instalação do Git

11

Sincronizando repositórios

14min

12

Para saber mais: Chave SSH

13

Para saber mais: O comando git remote

14

Conexão do repositório local com o Github

15

Faça como eu fiz: compartilhe seu projeto

16

Para saber mais: autenticação por token no Github

17

Desafio: hora da prática.

18

O que aprendemos?

OUTROS LINKS

Discord Alura

Fórum do curso

Voltar para Dashboard

MODO CLARO

Nicolas Sartor Parisotto

11.2k xp

Vimos no vídeo anterior que para fazer o link entre o repositório local, que está em nosso computador, com o repositório remoto, que está no GitHub, devemos utilizar o comando `git remote`.

Esse comando tem algumas variações e parâmetros opcionais que podem ser úteis em certas situações. Confira a seguir exemplos de uso desse comando:

1 - Adicionar um repositório remoto:

Quando você deseja estabelecer uma conexão entre seu repositório local e um repositório remoto, como aquele hospedado no GitHub, o comando `git remote add` é a ferramenta essencial. Essa etapa é crucial para possibilitar a colaboração e o compartilhamento de código com outras pessoas, bem como para fazer backup de seu trabalho em um servidor remoto.

A sintaxe básica do comando é a seguinte:

```
git remote add apelido url
```

COPIAR CÓDIGO

'apelido': Este é um nome que você atribui ao repositório remoto. Geralmente, se utiliza nomes descritivos, como "origin" para o repositório principal no GitHub, mas você pode escolher nomes que façam sentido para o seu projeto.

'url': Aqui, você insere a URL do repositório remoto. Esta URL é única para cada repositório remoto e serve como o endereço para acessar e interagir com ele pela internet. Certifique-se de copiar a URL correta do repositório que deseja adicionar.

2 - Listar os repositórios remotos:

Para listar os repositórios remotos associados ao seu projeto local, você pode utilizar o comando `git remote -v`. Isso exibirá uma lista de todos os repositórios remotos vinculados ao seu projeto, juntamente com suas URLs. Veja um exemplo:

```
git remote -v
```

COPIAR CÓDIGO

A saída será algo semelhante a:

```
origin https://github.com/seu-usuario/seu-projeto.git (fetch)
origin https://github.com/seu-usuario/seu-projeto.git (push)
```

COPIAR CÓDIGO

Essa lista é útil para verificar se os repositórios remotos estão configurados corretamente. Obs: Vai aparecer duplicado mesmo, pois o Git separa a url de envio de commits (push) da url de baixar commits (fetch).

3 - Remover um repositório remoto:

Caso você não precise mais de um repositório remoto específico, pode removê-lo com o comando `git remote remove apelido`. Substitua 'apelido' pelo nome do repositório remoto que deseja remover. Aqui está um exemplo:

```
git remote remove origin
```

COPIAR CÓDIGO

Após a execução deste comando, o repositório remoto chamado "origin" será desvinculado do seu projeto local.

4 - Alterar a URL de um repositório remoto:

Às vezes, é necessário atualizar a URL de um repositório remoto, como quando a URL do servidor do GitHub é modificada ou quando você copiou a URL incorreta ao adicionar o repositório remoto. Use o comando `git remote set-url apelido nova_url` para realizar essa atualização. Substitua 'apelido' pelo nome do repositório remoto e 'nova_url' pela nova URL que você deseja associar a ele. Aqui está um exemplo:

```
git remote set-url origin https://github.com/seu-usuario/seu-repo.git
```

COPIAR CÓDIGO

Isso atualizará a URL do repositório remoto "origin" para a nova URL especificada.

5 - Alterar o apelido de um repositório remoto:

Se você deseja renomear um repositório remoto, use o comando `git remote rename apelido novo_apelido`. Substitua 'apelido' pelo nome atual do repositório remoto e 'novo_apelido' pelo novo nome que você deseja atribuir a ele. Aqui está um exemplo:

```
git remote rename origin novo-origin
```

COPIAR CÓDIGO

Isso renomeará o repositório remoto de "origin" para "novo-origin".

Lembre-se de que o comando git remote é fundamental para a gestão de conexões entre seu repositório local e repositórios remotos, permitindo a colaboração eficiente e o controle de versão. Praticar esses comandos em seu ambiente de desenvolvimento ajudará a consolidar seu entendimento.

