

Tutorial de Fluxo de Trabalho com Git

1. Clonando o Repositório

Para começar a trabalhar no projeto, você precisa clonar o repositório para sua máquina local.

```
git clone https://github.com/DaniloVitoria/LibraryApi.git
```

```
cd LibraryApi
```

2. Criando uma Nova Branch

Antes de começar a trabalhar em uma nova funcionalidade ou correção de bug, crie uma nova branch. É uma boa prática nomear a branch de forma descritiva.

```
git checkout -b nome-da-sua-branch
```

3. Fazendo Alterações

Realize as alterações necessárias no código. Após fazer as alterações, você pode verificar o status dos arquivos modificados.

```
git status
```

4. Adicionando Alterações

Adicione os arquivos que você deseja incluir no próximo commit. Você pode adicionar arquivos específicos ou todos os arquivos modificados.

```
git add nome-do-arquivo
```

ou para adicionar todos os arquivos

```
git add .
```

5. Fazendo um Commit

Depois de adicionar as alterações, faça um commit com uma mensagem descritiva.

```
git commit -m "Descrição clara do que foi alterado"
```

6. Sincronizando com o Repositório Remoto

Antes de enviar suas alterações, é uma boa prática garantir que sua branch esteja atualizada com a branch principal (branch usada no repositório remoto `main`).

```
1 git checkout main
```

```
2 git pull origin main
```

```
3 git checkout nome-da-sua-branch
```

```
4 git merge main
```

7. Resolvendo Conflitos (se necessário)

Se houver conflitos durante o merge, você precisará resolvê-los manualmente. Após resolver os conflitos, adicione os arquivos novamente e faça um commit.

```
1 git add nome-do-arquivo-resolvido
```

```
2 git commit -m "Resolvendo conflitos"
```

8. Enviando Alterações para o Repositório Remoto

Depois de fazer o commit, você pode enviar suas alterações para o repositório remoto.

```
git push origin nome-da-sua-branch
```

9. Criando um Pull Request

Após enviar suas alterações, vá até o repositório no GitHub e crie um Pull Request (PR) para que suas alterações sejam revisadas e mescladas na branch principal.

10. Atualizando sua Branch com Mudanças do Repositório Remoto

Se outras pessoas fizerem alterações na branch principal enquanto você está trabalhando, você pode precisar atualizar sua branch novamente.

```
1 git checkout main  
2 git pull origin main  
3 git checkout nome-da-sua-branch
```

```
4 git merge main
```

11. Excluindo uma Branch Local

Depois que suas alterações forem mescladas, você pode excluir a branch local que não é mais necessária.

```
1 git branch -d nome-da-sua-branch
```