

Aula 04

VERSIONAMENTO DE CÓDIGO - GIT



DIGITAL

Professor
Kauê Maia

O que vamos aprender nesta aula?

- Ver o histórico de commits
- Recuperar um ponto do repositório
- Ignorar determinados arquivos

Lembre de ver a documentação: <https://git-scm.com/docs>

Vendo histórico de commits:

git log

git log --pretty=oneline

Organizar em 1 linha

git log --stat

git log -p

Mais informações

Ver as modificações

git log -p -2

Ver as 2 últimas modificações

git log --since=2.days

Filtrar por tempo

git log --author='Kauê Maia'

Filtrar por autor

Recuperar um ponto do histórico:

Voltando 1 commit:

```
git reset --soft HEAD~  
git reset --hard HEAD~
```

Arquivos ficam no 'unstaged'
Volta exatamente para o snapshot

Voltando para um commit específico:

```
git reset --soft [hash]  
git reset --hard [hash]
```

Lembre de ver antes as diferenças:

```
git log --pretty=oneline -p -2
```

Por exemplo...

Que HEAD (cabeça) foi esse?

O HEAD está sempre apontando para o commit que estamos usando.

Quando passamos como parâmetro de um comando **HEAD~** estamos dizendo que queremos apontar para o commit anterior ao que estamos.

Podemos passar um depois do ~(til) para dizer quantas gerações queremos navegar ou usar N vezes o ~(til):

Ex.:

git checkout HEAD~2

Voltando 2 commits

git checkout HEAD~~

Voltando 2 commits também

git checkout master

Para voltar a nossa branch padrão

No comando git log vemos onde o HEAD está apontando:

git log --pretty=oneline

Por exemplo...

Ignorando arquivos:

Para o Git ignorar arquivos ou pastas, você deve criar um arquivo .gitignore e adicionar nele linha a linha arquivos e pastas a serem ignoradas.

Exemplos:

```
echo 'vscode/' > .gitignore
```

```
echo 'pessoal.txt' >> .gitignore
```


MATONDO!

(OBRIGADO)



Diversidade para Inovação.

Acesse:
<https://toti.site/>