

Ver log simplificado de todos os branches com gráfico

Mostra os commits em uma linha, em forma de gráfico, incluindo todos os branches.

```
git log --oneline --graph --all
```

Filtrar commits por autor

Exibe somente commits feitos por um autor específico.

```
git log --author="seu_nome"
```

Filtrar commits por data

Mostra os commits realizados nos últimos 14 dias.

```
git log --since="2 weeks ago"
```

Ver alterações feitas entre arquivos, commits ou branches

Compara alterações não commitadas ou entre branches.

```
git diff
```

```
git diff main..dev
```

Ver quem modificou cada linha de um arquivo

Mostra o autor, data e hash do commit responsável por cada linha.

```
git blame nome_do_arquivo
```

Recuperação e desfazer ações

Ver o hash dos commits rapidamente

Lista os commits de forma resumida, com seus respectivos hashes.

```
git log --oneline
```

Voltar para um commit específico sem mudar a branch atual

Usado para visualizar o estado de um commit específico.

```
git checkout e4f5g6h
```

Resetar a branch para um commit específico (destrutivo)

Altera o histórico da branch atual para um commit anterior.

```
git reset --hard e4f5g6h
```

Forçar envio da branch resetada para o GitHub

Atualiza o repositório remoto com o novo estado da branch.

```
git push --force origin sua-branch
```

Desfazer o último commit e manter arquivos

Remove o último commit, mas mantém os arquivos prontos para novo commit.

```
git reset --soft HEAD~1
```

Desfazer o último commit e remover arquivos

Retira o commit e os arquivos do stage, mas mantém as alterações locais.

```
git reset --mixed HEAD~1
```

Desfazer o último commit e as alterações feitas (perigoso)

Remove o commit, o stage e apaga as alterações locais.

```
git reset --hard HEAD~1
```

Desfazer um commit sem apagar o histórico

Cria um novo commit que desfaz as mudanças do commit anterior.

```
git revert abc123
```

Ver histórico de referências ocultas e ações antigas

Mostra o histórico completo de alterações, incluindo resets e checkouts.

```
git reflog
```

Restaurar o repositório para um ponto anterior no reflog

Permite voltar ao estado do repositório em uma posição do reflog.

```
git reset --hard HEAD@{2}
```