



DevMasterTeam

## Git log - Variações

A execução do comando `git log` pode ficar confusa quando existem muitos registros a serem mostrados. Felizmente, esse comando possui algumas variações que podem facilitar a visualização.

`log log -n`

Faz a listagem de logs limitado a N registros.

`git log -p`

Mostra as mudanças de cada arquivo de acordo com o commit. É como se fosse feito um git diff em cada commit.

`git log --stat`

Mostra a quantidade de alterações em cada arquivo em formato visual, sem mostrar as mudanças específicas.

`git log --shortstat`

Mostra somente a quantidade de alteração de forma menos visual que `--stat`.

`git log --name-only`

Mostra a lista de arquivos modificados em cada commit.

`git log --name-status`

Mostra a lista de arquivos modificados em cada commit e para cada arquivo mostra as letras M, A ou D.

M - Arquivo modificado

A - Arquivo adicionado

D - Arquivo removido

`git log --abbrev-commit`

Faz exatamente igual ao comando `git log`, porém mostra somente poucos caracteres do commit, ao invés de todos os 40.

`git log --relative-date`

Mostra os commits com a data relativa comparado a data do dia atual. Ao invés de mostrar a data em que foi feita, mostra quanto tempo atrás foi feito.

Por exemplo,

"8 semanas atrás", "2 dias atrás"

`git log --graph`

Mostra a lista de commits juntamente com alguns caracteres no início da linha que mostram quando uma branch surgiu e quando houve um merge.

## git log --oneline

Dada a quantidade de variações e customizações que são possíveis de ser feitas no comando `git log --oneline`, a seção abaixo trata somente destas variações.

`git log --oneline`

Faz a listagem de commits mostrando o hash abreviado e a mensagem usada para o commit. Além disso, como o nome já diz, cada commit é mostrado em somente uma linha.

Também é possível informar o seu próprio formato através do parâmetro `--pretty:.format`. Abaixo, algumas das configurações disponíveis para serem usadas.

`%H`

Mostra o hash do commit.

`%h`

Mostra o hash do commit abreviado.

`%P`

Mostra o hash do commit anterior.

`%p`

Mostra o hash do commit do commit anterior abreviado.

`%an`

Mostra o nome do autor do commit.

```
%ae
```

Mostra o e-mail do autor do commit.

```
%ar
```

Mostra a data relativa ao dia de hoje. Há quanto tempo atrás a alteração foi feita.

```
%s
```

Mostra o comentário realizado no commit.

Abaixo, alguns exemplos das configurações acima sendo utilizadas.

```
git log --pretty=format:"%h - %p: Feito por %an"
```

```
git log --pretty=format:"%h: %ar - (%ae) %an"
```

O comando git log --oneline pode ser escrito da seguinte maneira:

```
git log --pretty=format:"%h %s"
```