

Customizando git log

Entretanto, não temos uma visão clara do histórico dos commits dessa forma. Muitas vezes queremos formatar o log para analisar como as branches se juntam, qual a linha do tempo entre elas, entre outros.

O comando `git log` suporta várias customizações para a saída. Algumas opções úteis são:

--oneline : Condensa o histórico de commits em uma linha por commit

--color : Tenta colorir o output para um resultado mais agradável

--graph : Monta o gráfico das branches em modo texto

--abbrev-commit : Abrevia o hash do commit. A flag do **--oneline** já faz isso

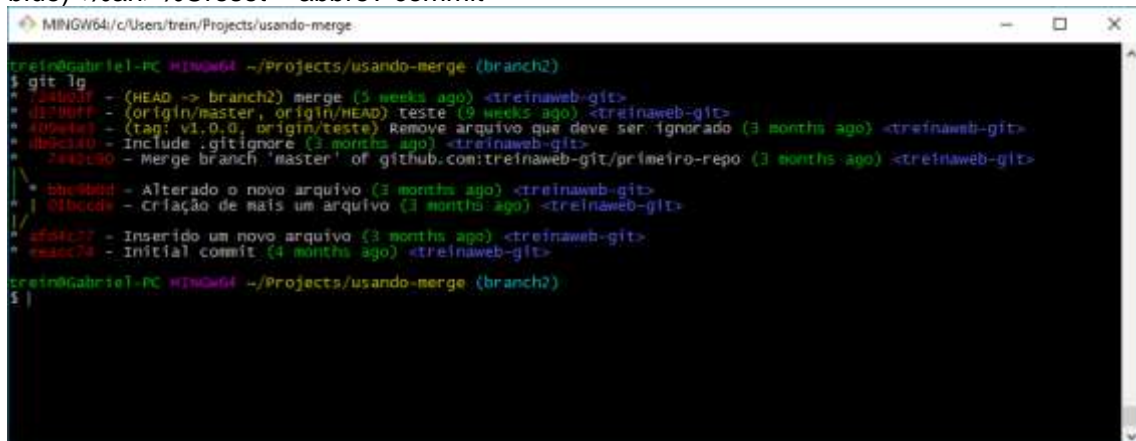
--pretty : Permite que sejam utilizados placeholders para as informações dos commits. A lista completa de opções você encontra ao executar `git log --help`

Combinando algumas dessas opções, podemos criar históricos como esse:



1

```
git log --color --graph --pretty=format:'%Cred%h%Creset -%C(yellow)%d%Creset %s
%Cgreen(%cr) %C(bold
blue)<%an>%Creset' --abbrev-commit
```



```
trein@Gabriel-PC MINGW64 ~/Projects/usando-merge (branch2)
$ git lg
* 7a42c90 - (HEAD -> branch2) merge (3 weeks ago) <treinaweb-git>
* 017b0ff - (origin/master, origin/HEAD) teste (9 weeks ago) <treinaweb-git>
* 409e4e1 - (tag: v1.0.0, origin/teste) Remove arquivo que deve ser ignorado (3 months ago) <treinaweb-git>
* 0b6c340 - Include .gitignore (3 months ago) <treinaweb-git>
* 7a42c90 - Merge branch 'master' of github.com:treinaweb-git/primeiro-repo (3 months ago) <treinaweb-git>
* 0b6c340 - Alterado o novo arquivo (3 months ago) <treinaweb-git>
* 017b0ff - Criação de mais um arquivo (3 months ago) <treinaweb-git>
* 0b6c340 - Inserido um novo arquivo (3 months ago) <treinaweb-git>
* 7a42c90 - Initial commit (4 months ago) <treinaweb-git>
trein@Gabriel-PC MINGW64 ~/Projects/usando-merge (branch2)
$ |
```

Se preferir, é possível definir apelidos para os comandos internos do `git`, assim temos um log customizado e de fácil execução:

