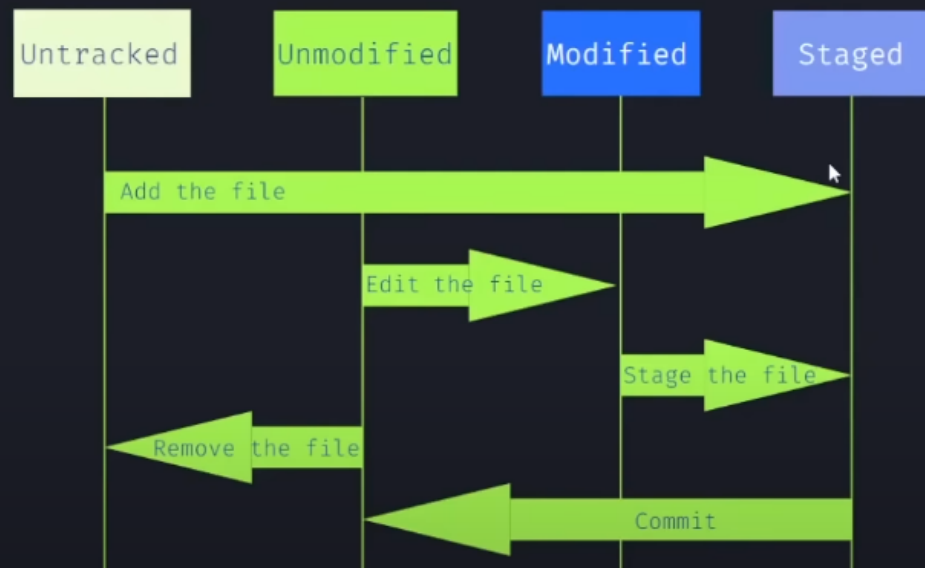


```
git -v #verificar se o git esta instalado e sua versao
git config --global user.name"" #configuracao de usuario
git config --global user.email"" #configuracao de email
git clone
https://github.com/GuilhermeSampaio/ModelagemConta.git
#clonar e baixar um repositorio ja existente no github via
HTTPS
git init #transformar a pasta / arquivo em um repositorio git
dentro da pasta
```

Estados do GIT



IS VÍDEOS

```
git status#informa o estado dos arquivos da pasta (se foram
modificados ou nao)
```

```
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    README.txt

nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT>
```

```
nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git add .\README.txt
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git status

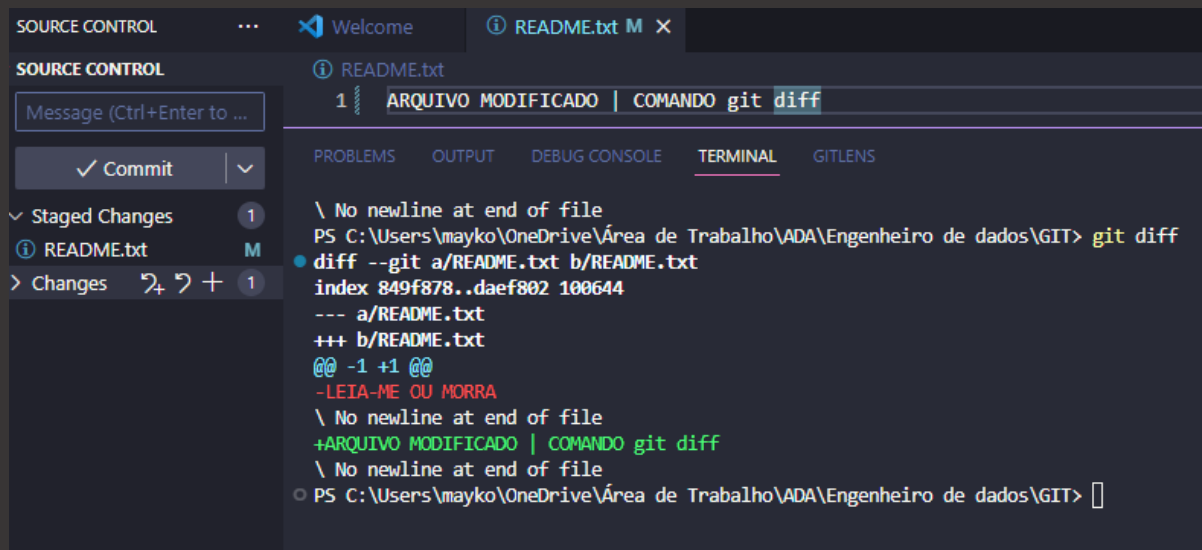
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   README.txt

PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT>
```

```
git diff #mostra as linhas em que o arquivo foi modificado
depois de comitado
```



```
1 ARQUIVO MODIFICADO | COMANDO git diff
```

Message (Ctrl+Enter to ...)

Commit

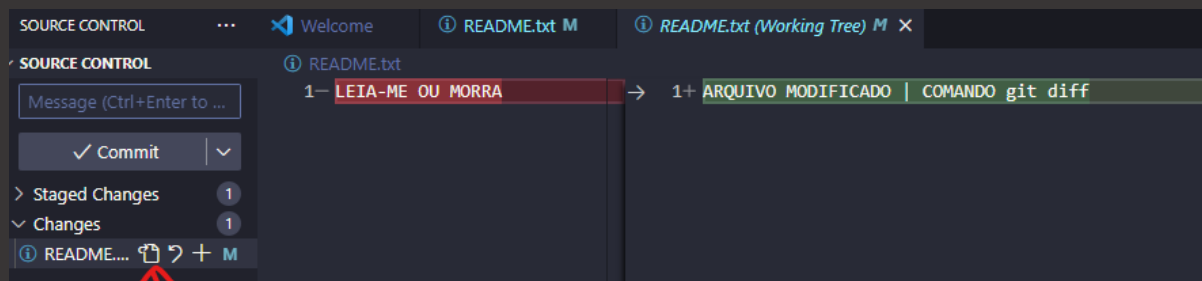
Staged Changes 1

Changes 1

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS

```
\ No newline at end of file
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git diff
diff --git a/README.txt b/README.txt
index 849f878..daef802 100644
--- a/README.txt
+++ b/README.txt
@@ -1,1 @@
-LEIA-ME OU MORRA
\ No newline at end of file
+ARQUIVO MODIFICADO | COMANDO git diff
\ No newline at end of file
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT>
```

OU



```
1- LEIA-ME OU MORRA → 1+ ARQUIVO MODIFICADO | COMANDO git diff
```

Message (Ctrl+Enter to ...)

Commit

Staged Changes 1

Changes 1

README... M

CLICK ALI

```
git commit -m "add new title" #comitando arquivo (dica: é legal comitar a cada alteracao do arquivo, mesmo se for pequena, como no exemplo: um novo titulo)
#COMMIT (salvar estado do projeto) SNAPSHOT
```

```
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git commit -m "add new title"
[master (root-commit) 6e103ee] add new title
1 file changed, 1 insertion(+)
create mode 100644 README.txt
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT>
```

```
git log #mostra o histórico de todos os commits
```

```
1 ARQUIVO MODIFICADO | COMANDO git diff
2 You, 4 minutes ago • Uncommitted changes
3 nova modificacao
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS

```
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git log
commit 28c2c6742d00a39b973e1413044a5da03b698eeb (HEAD -> master)
Author: maykonsf7 <maykondasilvaferreira1414@gmail.com>
Date: Fri May 26 15:53:13 2023 -0300

    nova modificacao

commit 6e103ee860f1de51b64305aa30bceada8606de8b
Author: maykonsf7 <maykondasilvaferreira1414@gmail.com>
Date: Fri May 26 14:54:52 2023 -0300

    add new title
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT>
```

welcome README.txt README.txt (working tree) ...

① README.txt

```
1 ARQUIVO MODIFICADO | COMANDO git diff
2
3 nova modificacao
4+
5+ uso do comando restore You, 39 sec
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS

```
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT>
```

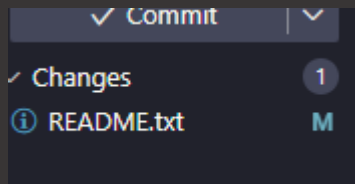
git restore .\README.txt #descarta as alteracoes feitas e volta ao arquivo cru nao modificado

▼ Staged Changes 1

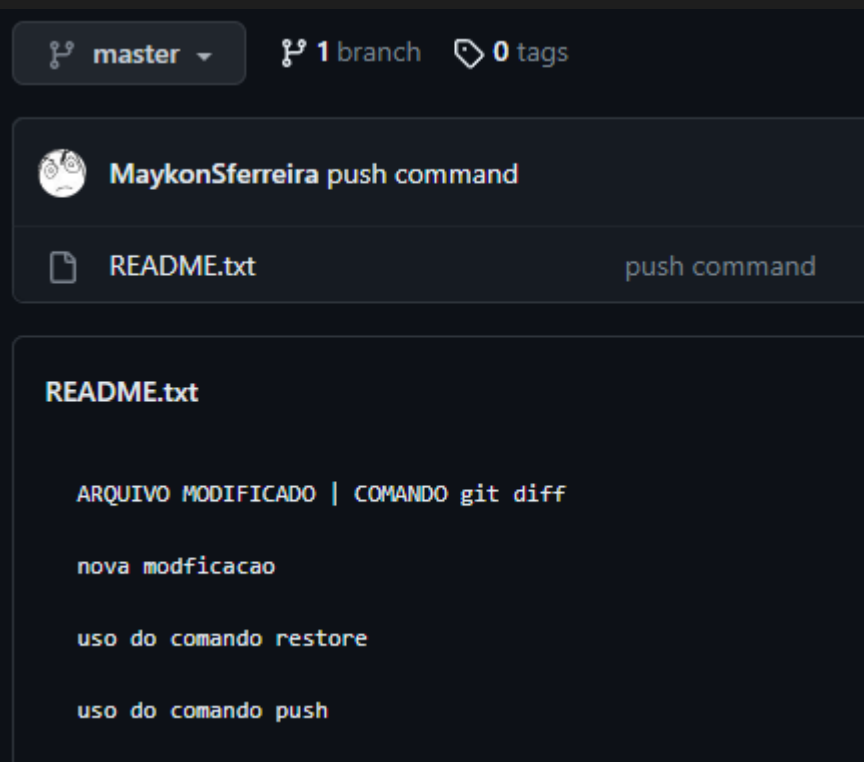
① README.txt M

▼ Changes 0

git restore --staged .\README.txt #usado quando se deseja retornar ao estado de apenas modificado (CHANGE) depois de comitado (add . STAGED)



```
git remote add origin  
https://github.com/MaykonSferreira/GIT-ADA.git #Adiciona uma  
conexao remota ao repositorio do github  
git remote #verifica o nome da conexao remota (ORIGIN)  
  
git push origin master #empurra o arquivo local (ORIGIN) para a  
branch principal (MASTER) do github
```



```
git pull origin master #puxa as alteracoes feitas na branch  
principal (MASTER) E ADICIONA para o meu repositorio local  
(ORIGIN) || Ex: QUANDO OUTRAS PESSOAS COMITAM O ARQUIVO DA  
BRANCH PRINCIPAL (MASTER)
```

```
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 672 bytes | 26.00 KiB/s, done.
From https://github.com/MaykonSferreira/GIT-ADA
   3af37fa..7901039 master    -> origin/master
There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.
```

```
git pull <remote> <branch>
```

If you wish to set tracking information for this branch you can do so with:

```
git branch --set-upstream-to=origin/<branch> master
```

```
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> []
```

You, 7 minutes ago | 1 author (You)

ARQUIVO MODIFICADO | COMANDO git diff

nova modificacao

uso do comando restore

uso do comando push You, 7 minutes ago

git log #mostra a localidade de onde estou nas ramificacoes
HEAD

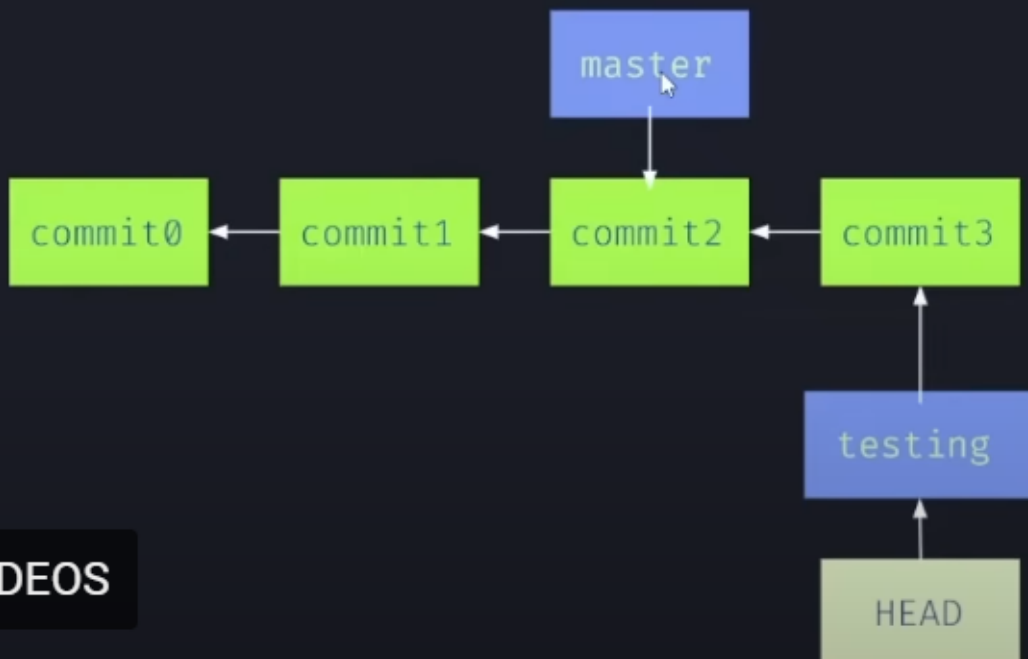
git log --oneline --decorate #mostra a localidade de onde esta
nas ramificacoes HEAD em uma linha(--oneline) e com
cores(--decorate)

```
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git log --oneline --decorate
28f1bd4 (HEAD -> master, origin/master) fetch command
7901039 pull command
3af37fa push command
28c2c67 nova modificacao
6e103ee add new title
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> []
```

```
git checkout testing #muda a localidade do HEAD para a branch
testing q eu criei
git log --oneline --decorate #mostra a nova localidade de onde
estou nas ramificacoes HEAD em uma linha(--oneline) e com
cores(--decorate)
```

```
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git branch testing
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git checkout testing
Switched to branch 'testing'
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git log --oneline --decorate
28f1bd4 (HEAD -> testing, origin/master, master) fetch command
7901039 pull command
3af37fa push command
28c2c67 nova modificacao
6e103ee add new title
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> []
```

Branches

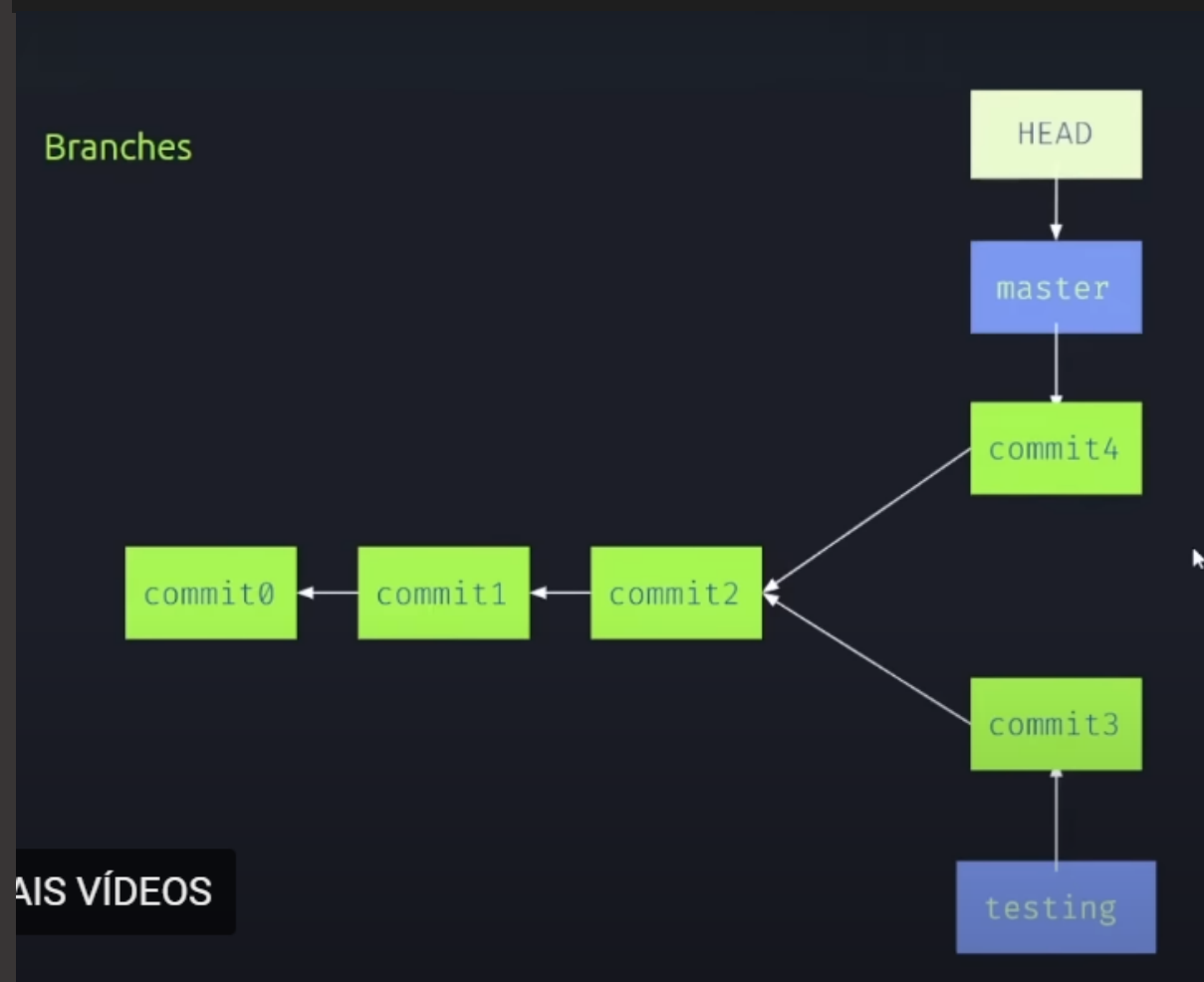


IS VÍDEOS

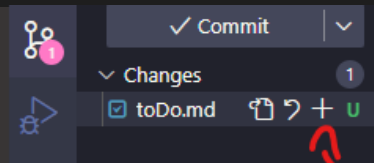
```
COMITEI UM NOVO ARQUIVO NA BRANCH TESTING
DEPOIS MUDEI PARA A BRANCH MASTER
```

O NOVO ARQUIVO CRIADO NA BRANCH TESTING NAO APARECE NA BRANCH MASTER PERMANECENDO NO ÚLTIMO COMMIT (PASSADO) .
VER A FIGURA ACIMA

O QUE OCORREU FOI UMA RAMIFICAÇÃO/DIVISÃO DOS COMMITS
COMUMENTE USADO PARA QUE VÁRIOS DEV TRABALHEM AO MESMO TEMPO
EM UM PROJETO EM DIFERENTES PARTES DELE
EM UM CERTO PONTO DO DESENVOLVIMENTO ESSAS RAMIFICAÇÕES
PODERAO SE ENCONTRARAO NA LINHA PRINCIPAL DO CÓDIGO (COMMITTS)
VER A FIGURA ABAIXO



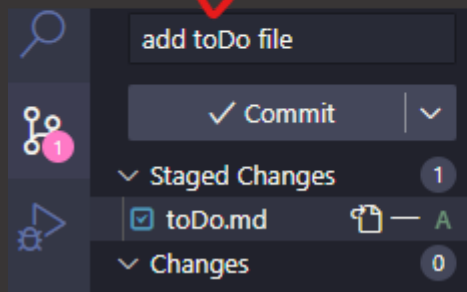
ADICIONAR ARQUIVO PARA STAGED ATRAVÉS DA EXTENSAO GIT LENS



ADD STAGE FILE

COMMITAR ARQUIVO ATRAVÉS DA EXTENSAO GIT LENS

COMMIT FILE

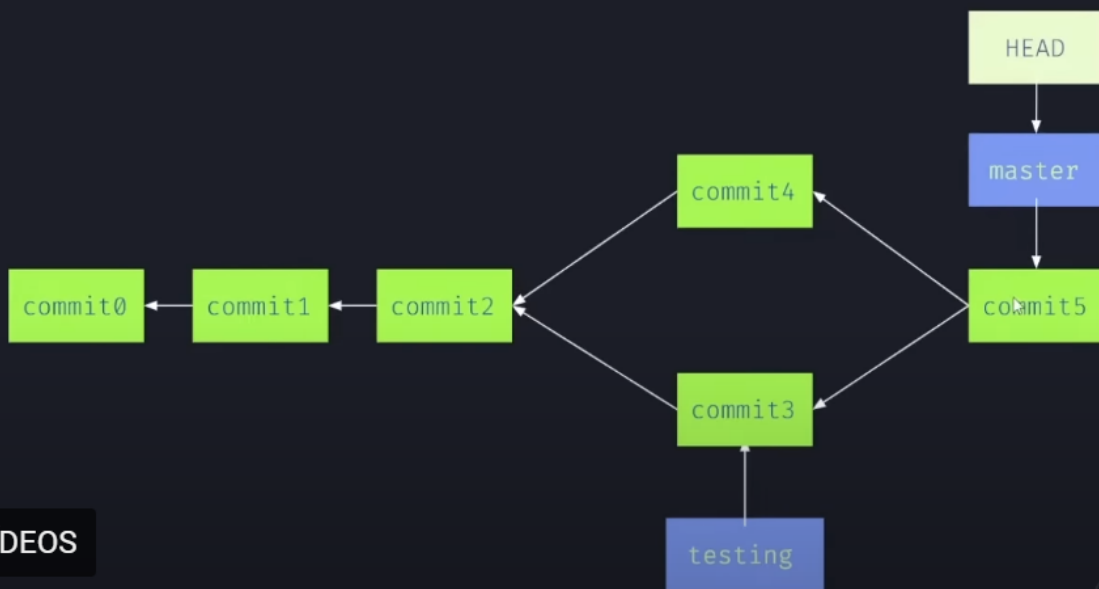


```
git merge testing #as alteracoes feitas na branch (TESTING) É FUNDIDA com o meu repositario (MASTER)
```

```
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> git merge testing
Updating 28f1bd4..a000a92
Fast-forward
 .gitignore | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 .gitignore
PS C:\Users\mayko\OneDrive\Área de Trabalho\ADA\Engenheiro de dados\GIT> 
```

VER A FIGURA ABAIXO

Branches



IS VÍDEOS