**Exercício de versionamento com Git**

**Nome:** Pedro Augusto Fernandes

**Matrícula:** 288736

**Data:** 13/03/2020

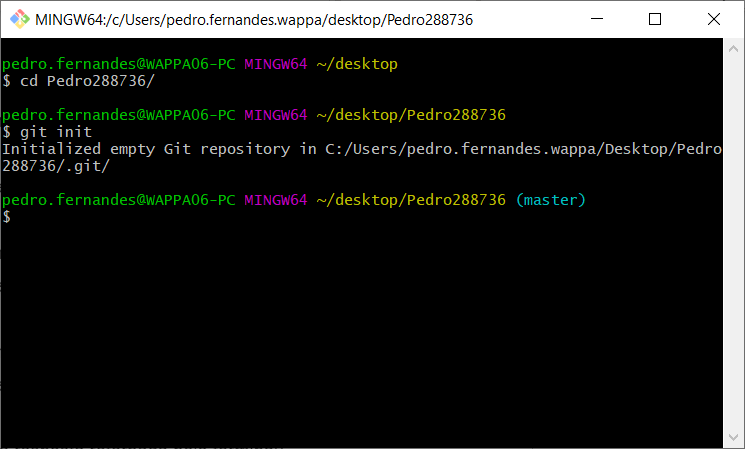
**Instruções**

1. Crie um diretório com o seu primeiro nome e seu número de matricula.

(exemplo: isaac12345678)

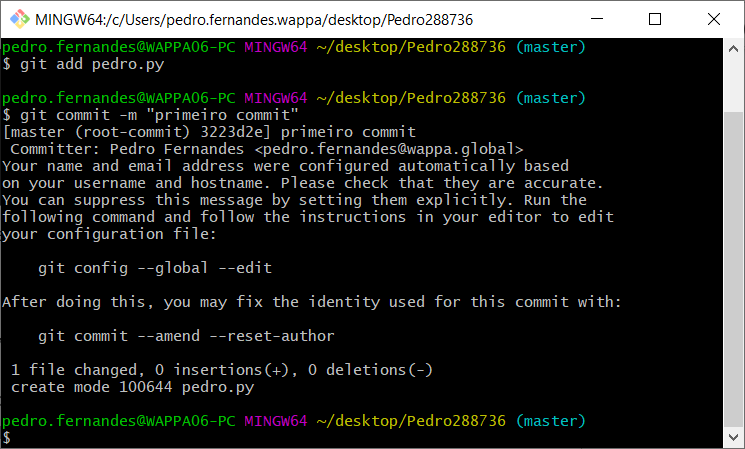
Entre no diretório que acabou de criar e inicie o versionamento com o comando **git init**.

(Cole aqui um *print* da tela do mostrando seu terminal)



2) Crie um arquivo em com o seu nome (exemplo: isaac.py), e inicie o rastreamento deste arquivo usando o comando **git add** e em seguida realize o commit desse arquivo com o comando **git commit**.

(Cole aqui o código do comando git, com a resposta retornada pelo terminal)



1. Altere o conteúdo do arquivo que você criou colocando um código que pede para o usuário o nome, e depois apresenta na saída o nome digitado.

Exemplo:

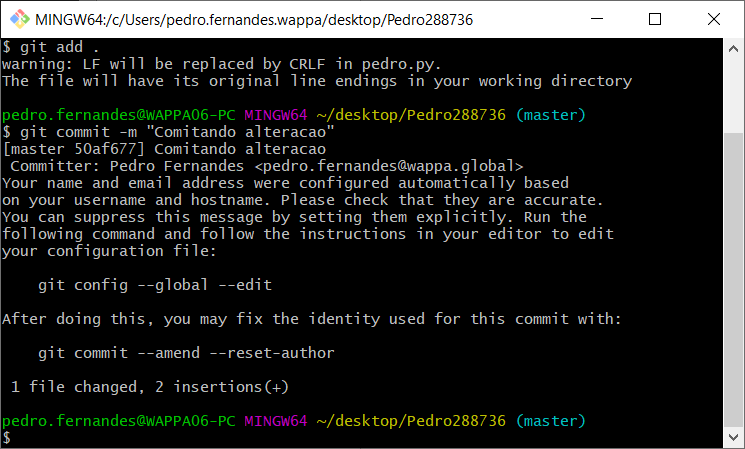
nome = input(“Digite seu nome: “)

print(“Seu nome é: “ + nome)

(Execute o programa e cole aqui um *print screen* do terminal)

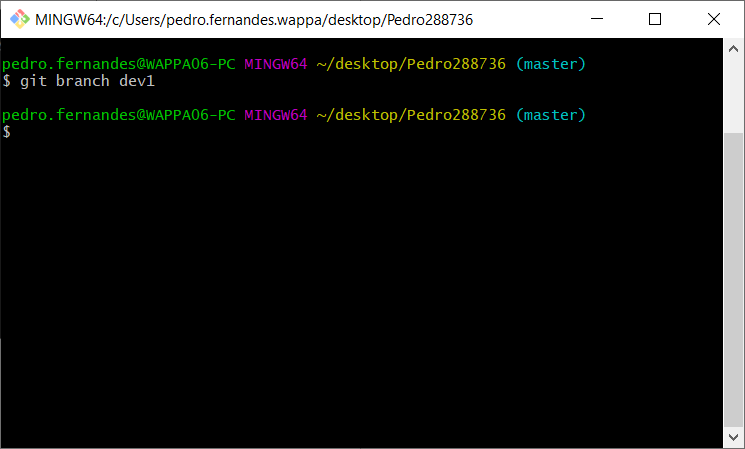
4) Realize o commit do projeto na branch master.

(Cole aqui o código do comando git, com a resposta retornada pelo terminal)

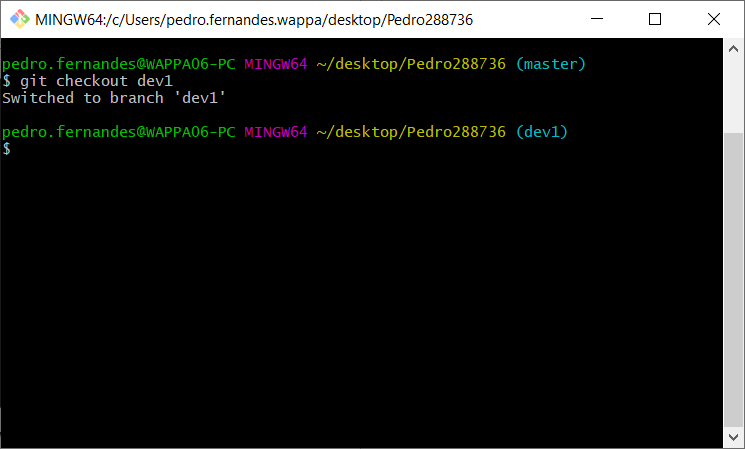


5) Crie uma nova branch chamada dev1 a partir da master.

(Cole aqui o código do comando git, com a resposta retornada pelo terminal)



6) Mude para sua nova branch com o comando: **git checkout dev1**.



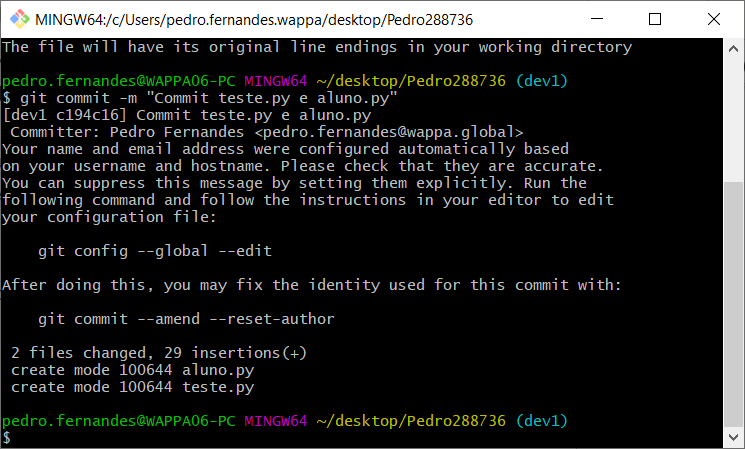
7) Na branch dev1 coloque os dois arquivos em python contendo uma classe chamada aluno.py e outro arquivo chamado teste.py. Dentro do arquivo teste.py use como teste o seu nome e seu número de matricula.

(Exemplo: O exemplo será disponibilizado pelo professor)

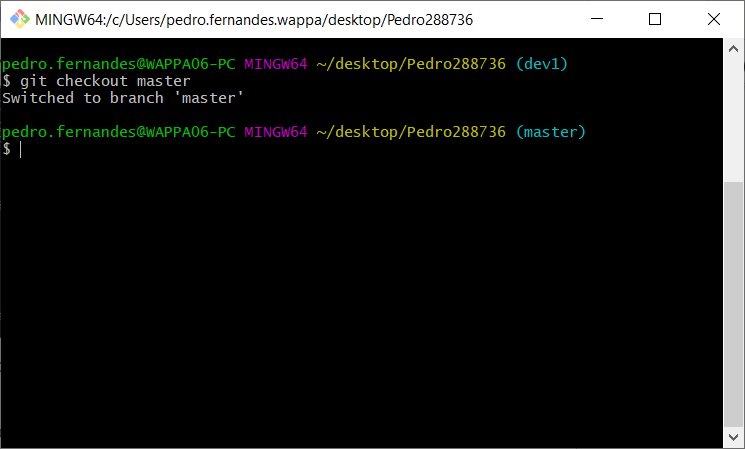
(Execute o programa teste.py e cole aqui um *print screen* do terminal, use como teste o seu nome e seu número de matricula)

8) Faça o commit na branch dev1.

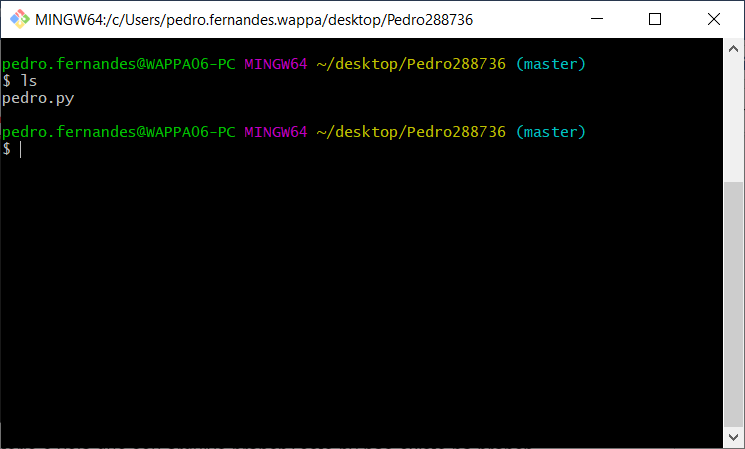
(Cole aqui o código do comando git, com a resposta retornada pelo terminal)



9) Mude para a branch master com o comando: **git checkout master**.



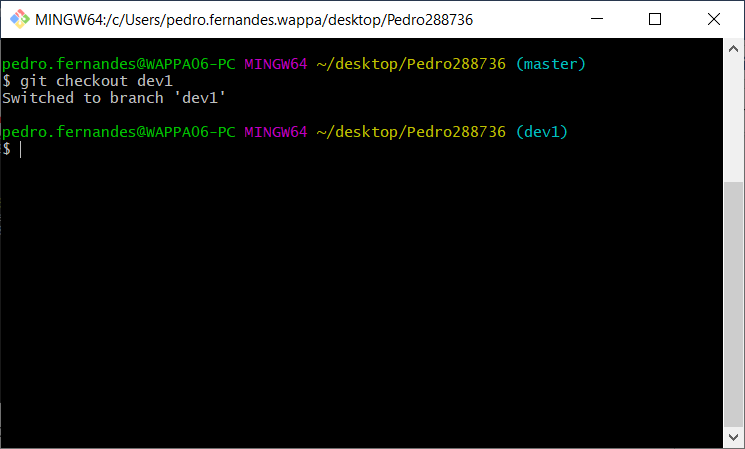
10) Agora verifique seu diretório e veja que seu arquivo branchTeste.py não existe na branch master.



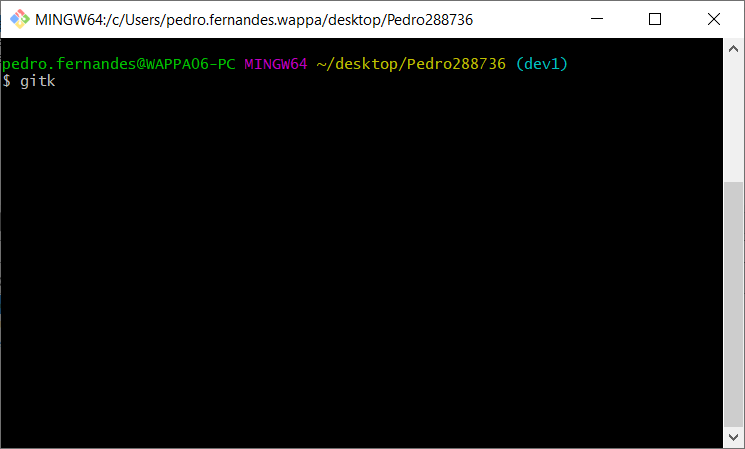
11) Digite o comando: **git log**, e veja que seu mais recente commit não está mais no log.



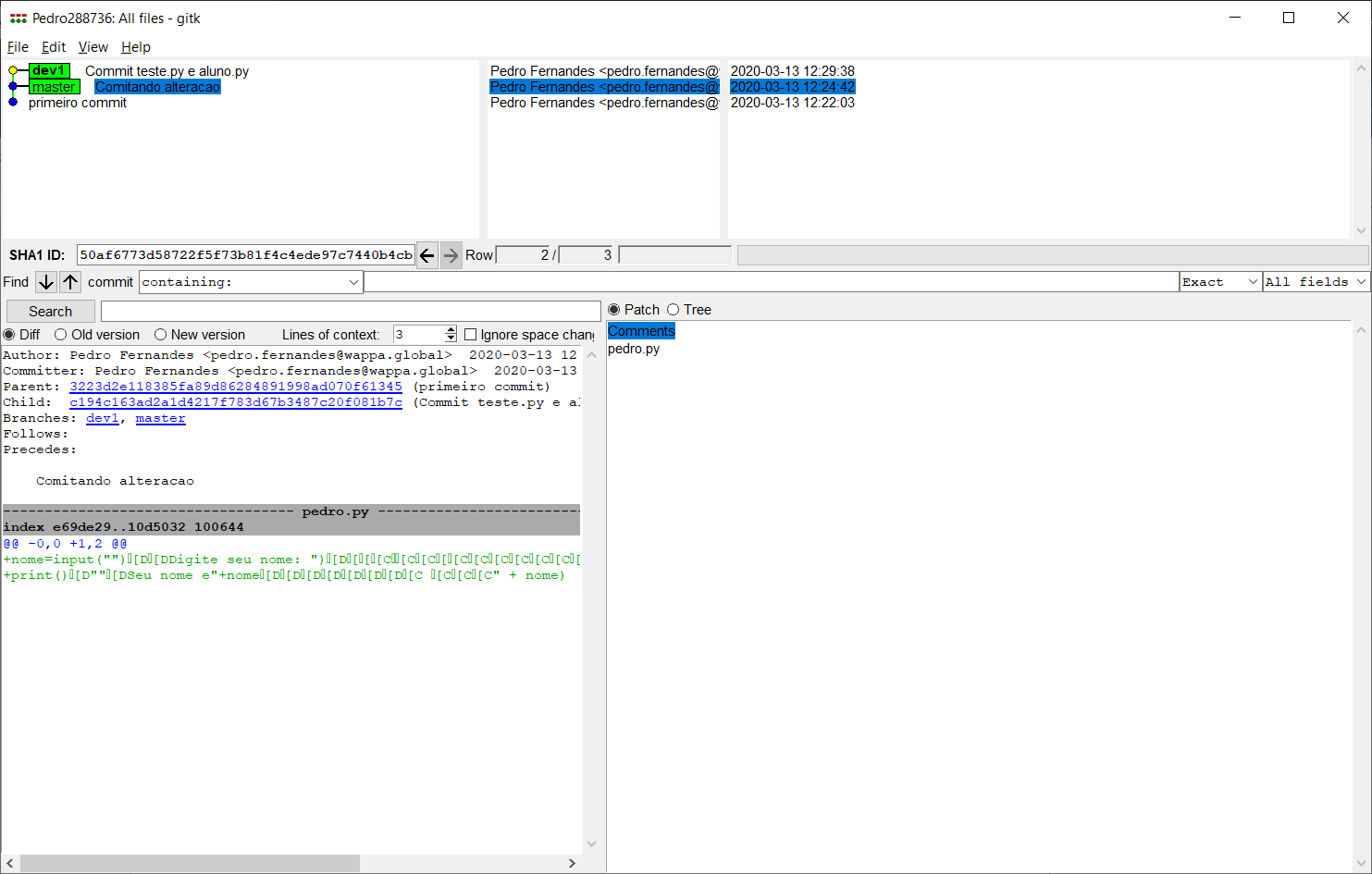
12) Mude para sua branch dev1 com o comando: **git checkout dev1**



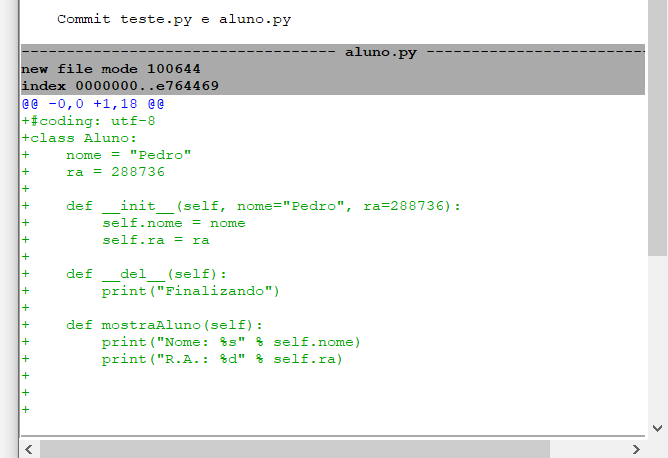
13) Digite o comando **gitk**



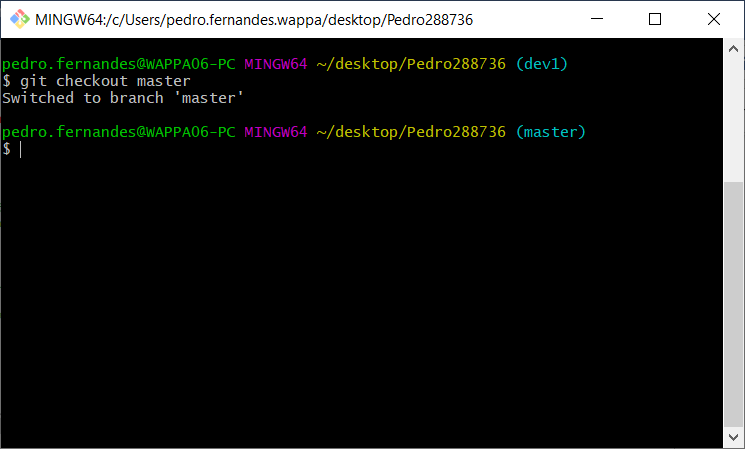
14) Veja que o programa **gitk** mostra que sua branch dev1 está com um commit na frente da master.



15) Com o programa gitk voçê também pode verificar as modificações que foram sendo realizadas em cada commit.

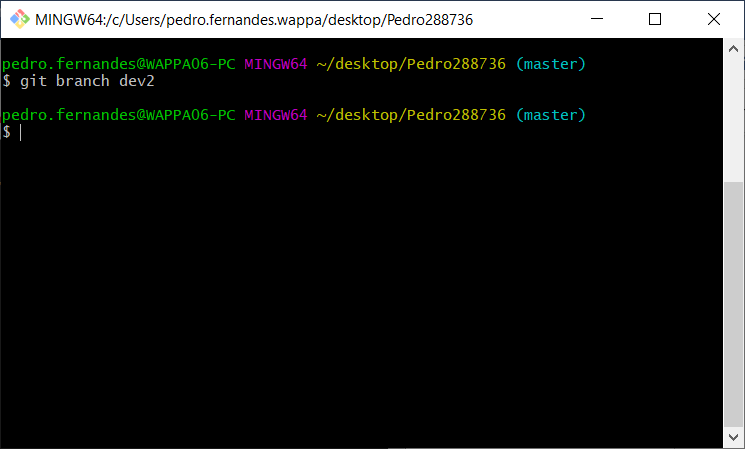


16) Mude para a branch master com o comando: **git checkout master**.



17) Crie uma nova branch chamada dev2 a partir da master.

(Cole aqui o código do comando git, com a resposta retornada pelo terminal)

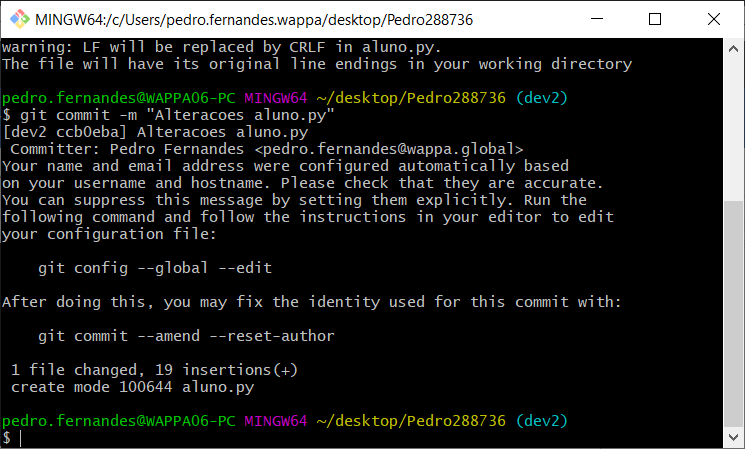


18) Na branch dev2 altere o arquivo aluno.py criando os atributos curso, inclua esse atributo nos parâmetros de entrada do construtor.

(Execute o programa teste.py e cole aqui um *print screen* do terminal, use como teste o seu nome, número de matricula e curso).

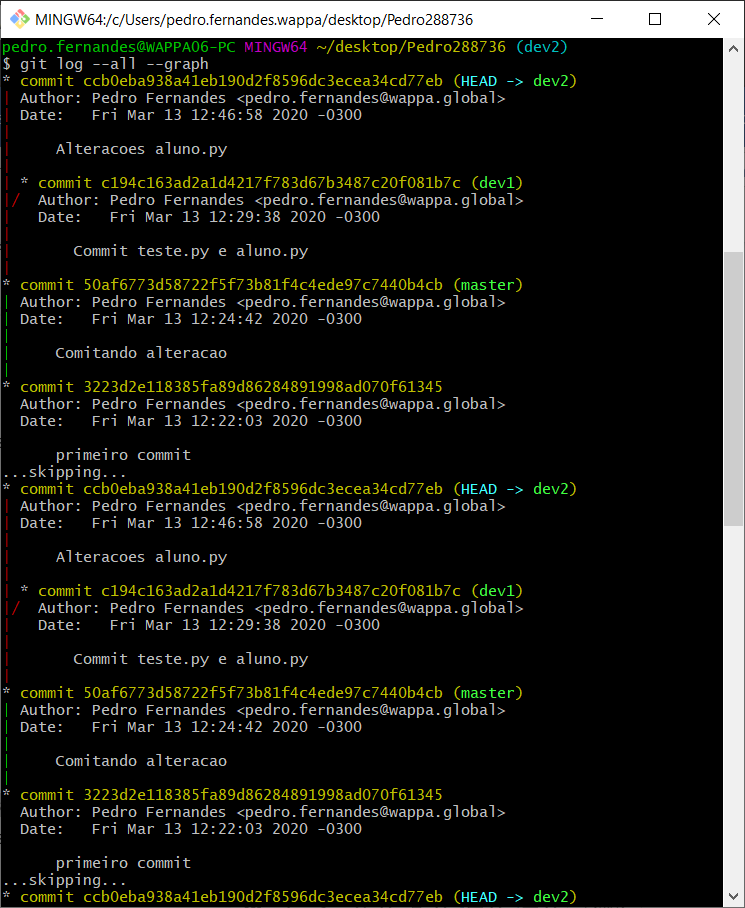
19) Faça o commit na branch dev2.

(Cole aqui o código do comando git, com a resposta retornada pelo terminal)

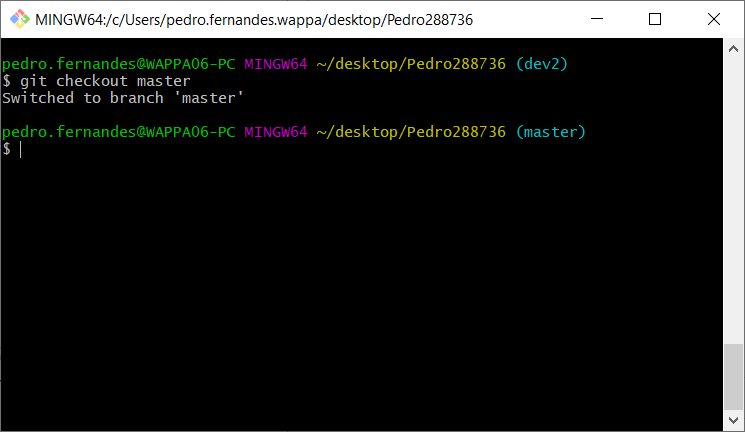


20) Execute o comando **git log –all –graph** para visualizar todos os commits do repositório..

(Cole aqui o código do comando git, com a resposta retornada pelo terminal)

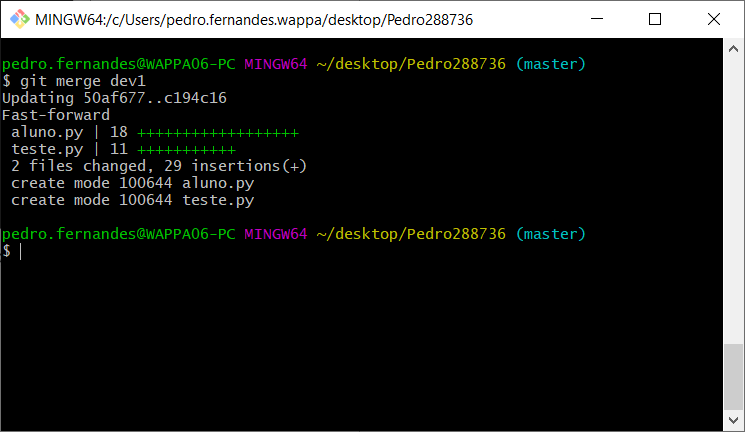


21) Mude para a branch master digitando o comando: **git checkout master**



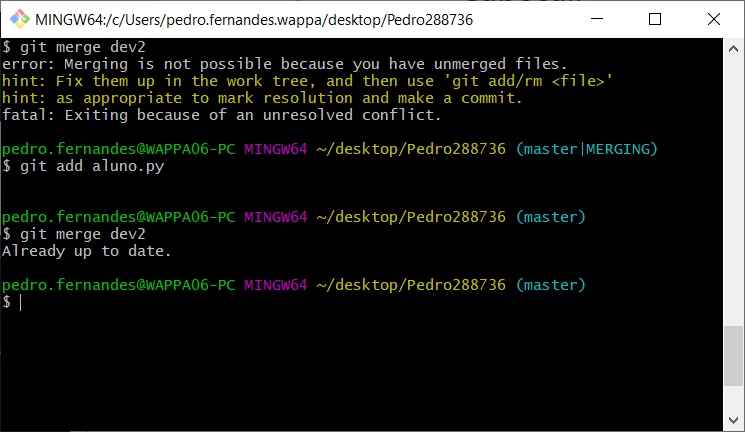
22) Agora vamos realizar o merge passando todas essas modificações da dev1 para a branch master com o comando: **git merge dev1**

(Cole aqui o código do comando git, com a resposta retornada pelo terminal)



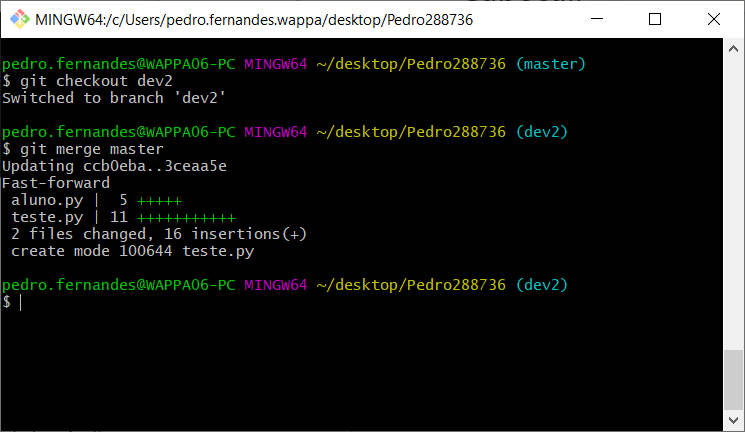
23) Faça o merge da dev2 com a master.

(Cole aqui o código do comando git)



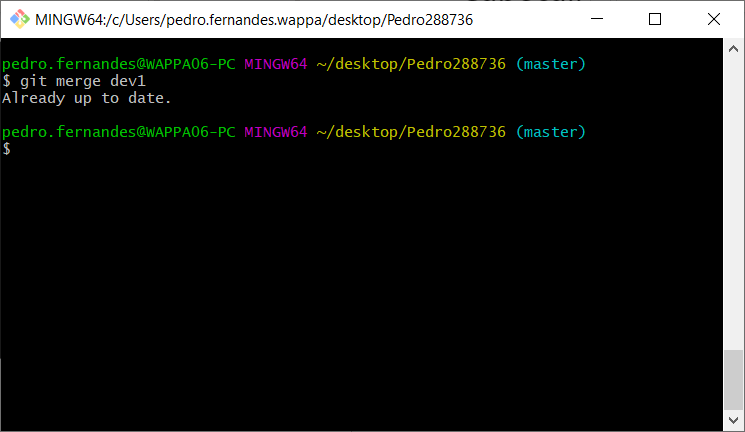
24) Faça o merge da master com a dev2.

(Cole aqui o código do comando git)



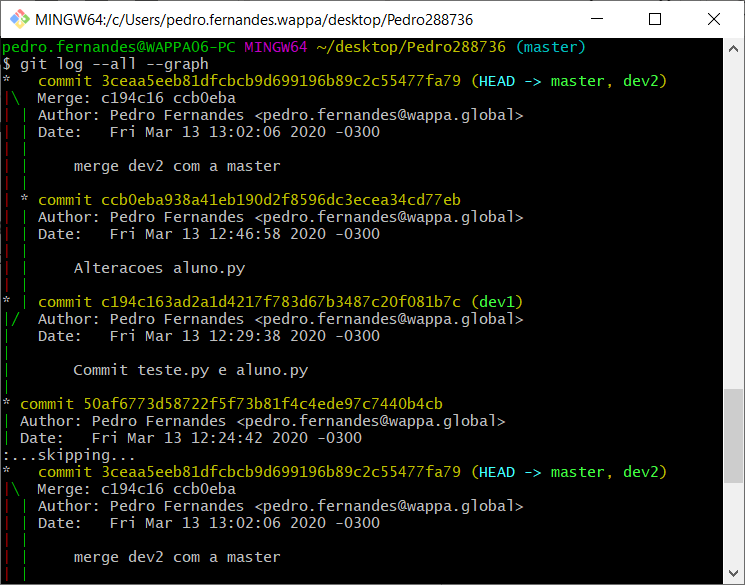
25) Faça o merge da dev1 com a master.

(Cole aqui o código do comando git)



26) Execute o comando **git log –all –graph** para visualizar todos os commits do repositório..

(Cole aqui o código do comando git, com a resposta retornada pelo terminal)



27) Na branch dev1 altere a classe aluno.py incluindo alguns comentários que explicam o que faz a classe e quais são os parâmetros de entrada do construtor.

(Cole aqui o código em python com os comentários)

#coding: utf-8

#Classe Aluno com atributos nome, ra e curso

class Aluno:

nome = "Pedro"

ra = 288736

curso="ads"

#construtor recebendo os parametros nome, ra e curso

def \_\_init\_\_(*self*, *nome*="Pedro", *ra*=288736, *curso*="ads"):

*self*.nome = nome

*self*.ra = ra

def \_\_del\_\_(*self*):

print("Finalizando")

#Mostra na tela o nome e o ra do aluno

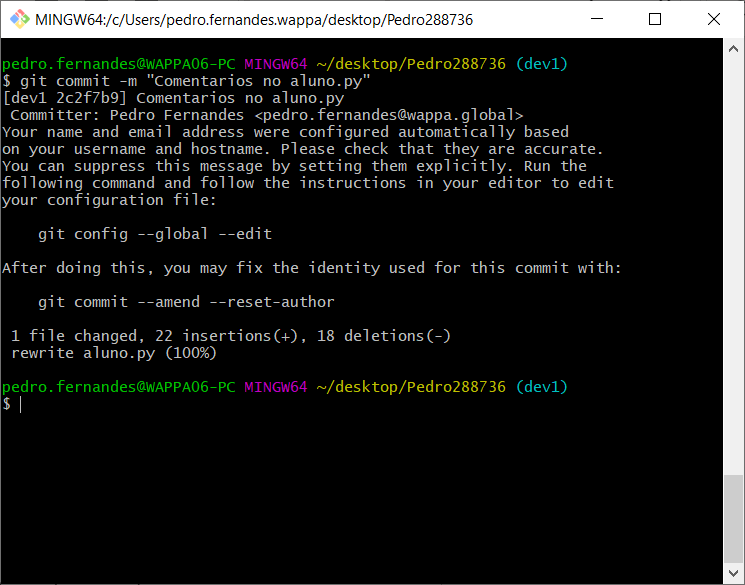
def mostraAluno(*self*):

print("Nome: %s" % *self*.nome)

print("R.A.: %d" % *self*.ra)

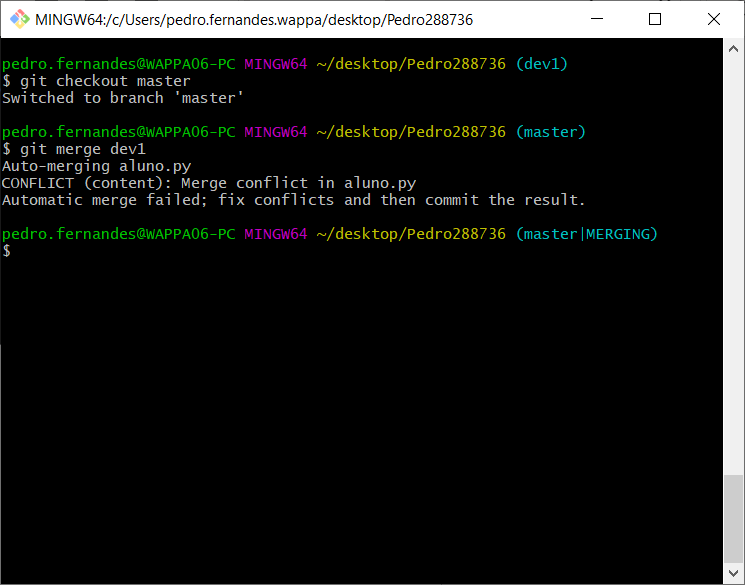
28) Faça o commit na dev1.

(Cole aqui o código do comando git)



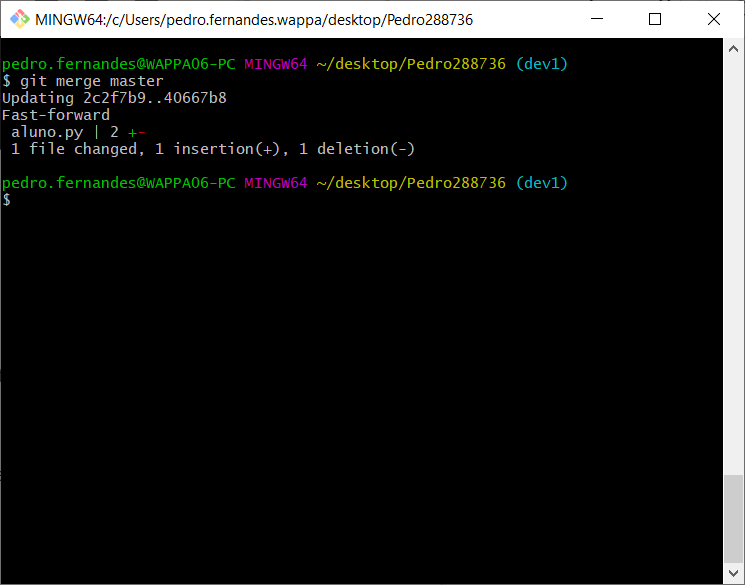
29) Faça o merge da dev1 com a master.

(Cole aqui o código do comando git)



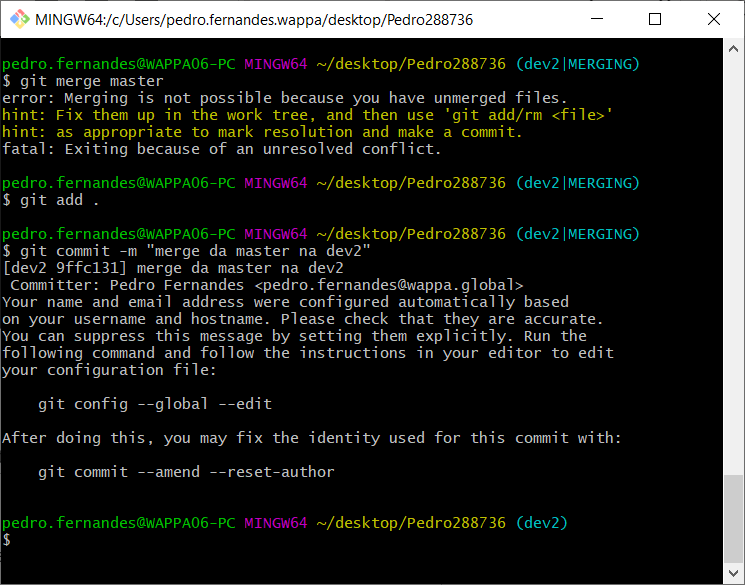
30) Faça o merge da master com a dev1.

(Cole aqui o código do comando git)



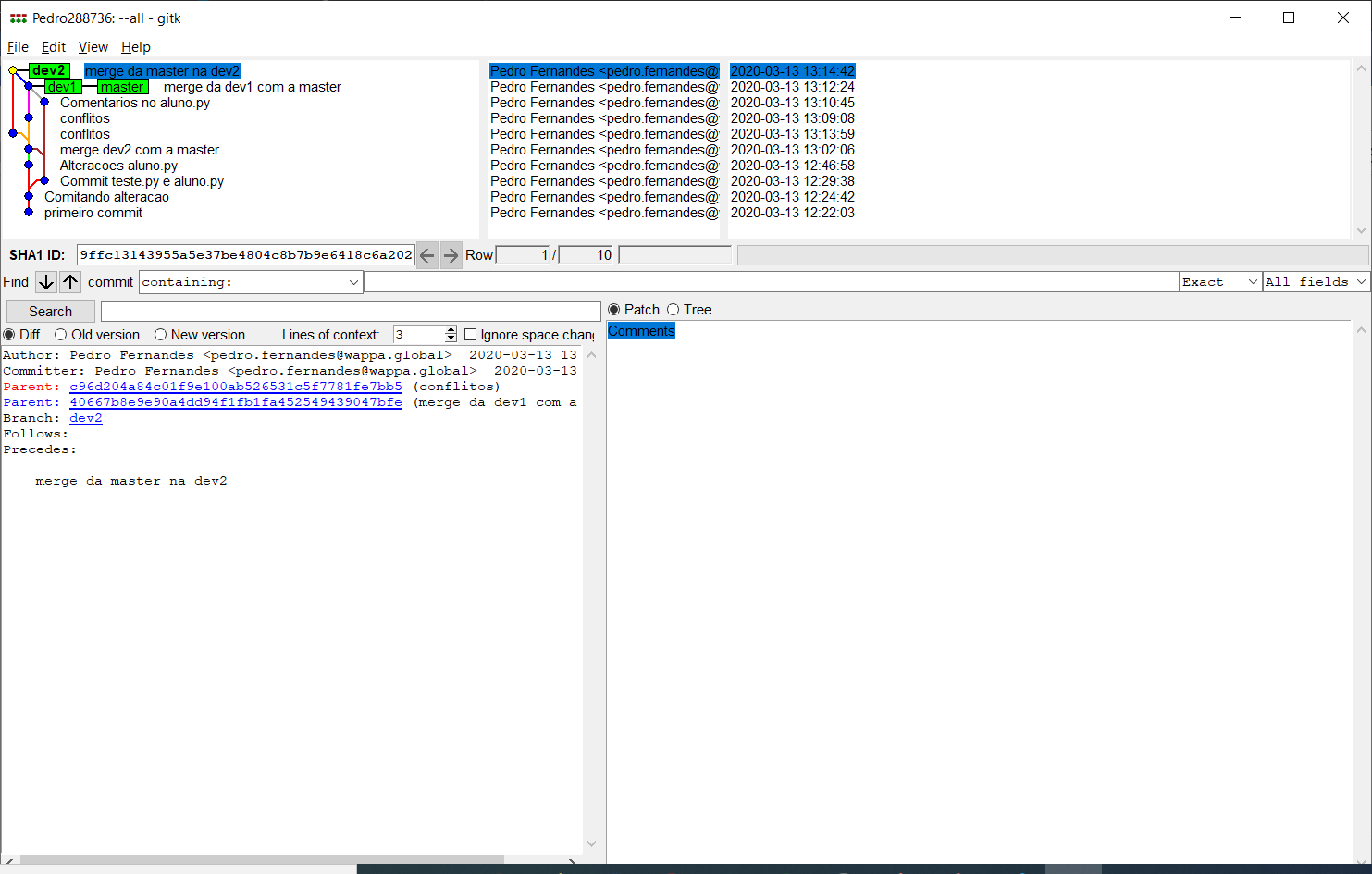
31) Faça o merge da master com a dev2.

(Cole aqui o código do comando git)



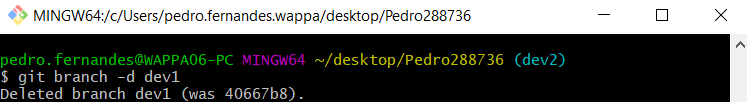
32) Execute o comando **gitk –all** para mostrar todo o histórico dos commits.

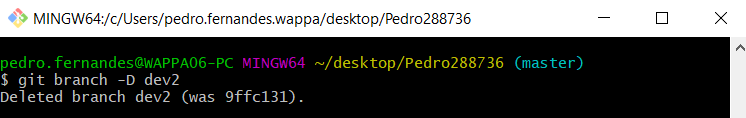
(Cole aqui o print da tela)



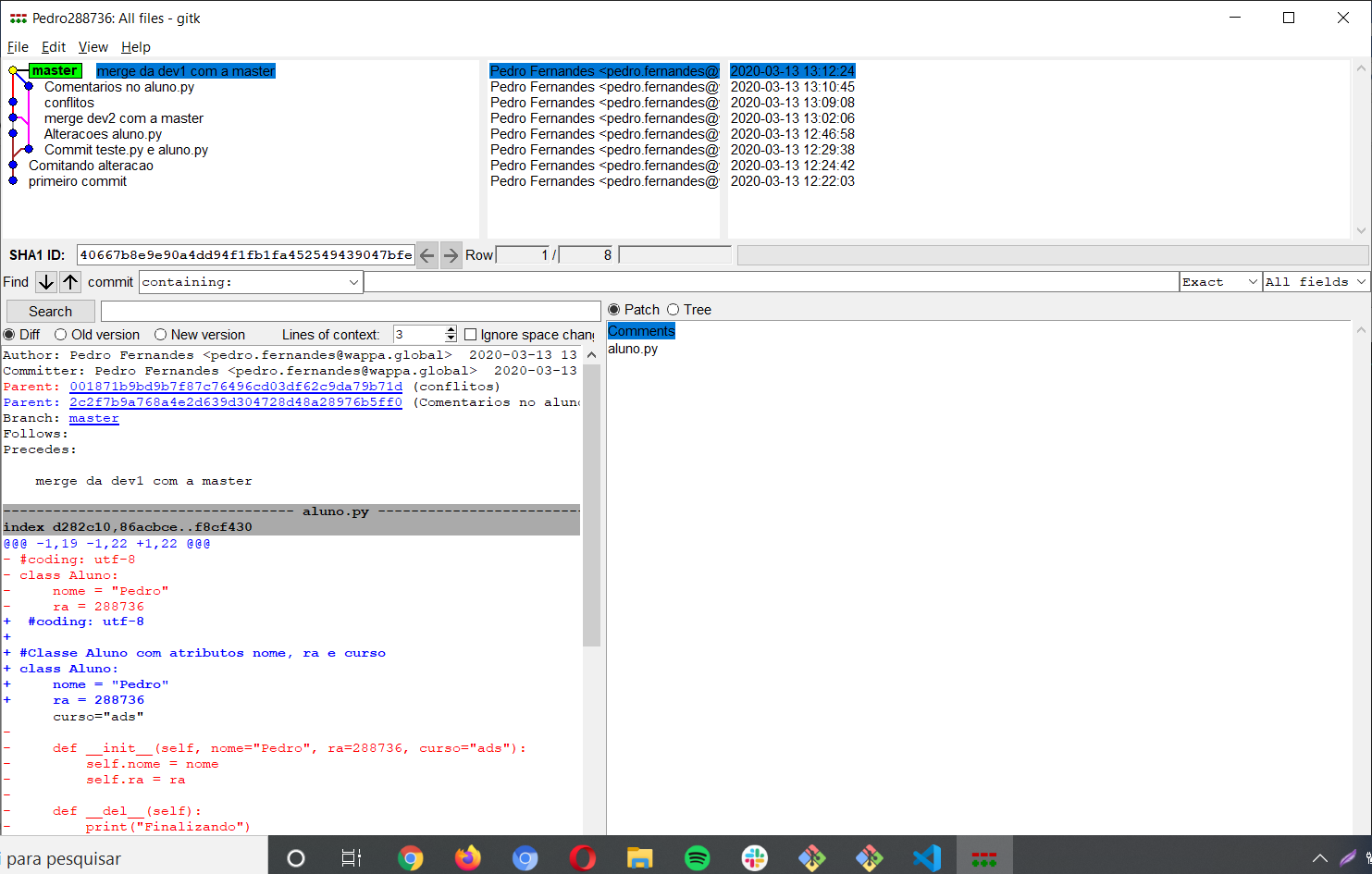
33) Apague as branches dev1 e dev2. Voçê pode remover com o comando: **git branch –D dev1**

(Cole aqui o código do comando git)





34) Agora digite o comando: **gitk**, e veja que sua branch não existe mais.



35) Execute o comando gitk –all para mostrar todo o histórico dos commits.

(Cole aqui o print da tela)

