

Nome: Danilo Marques da Silva

Matrícula: 2020111iinf0240

Atividade 10 – dia 26.10.2021

IMPORTANTE:

O OBJETIVO DESTA ATIVIDADE É VOCÊ EXECUTAR TODOS OS CÓDIGOS PROPOSTOS E TENTAR ENTENDER COMO ELE SE COMPORTA E PORQUÊ.

O CÓDIGO É QUASE O MESMO EM CADA UMA DAS QUESTÕES, MAS EM CADA QUESTÃO EXISTEM PEQUENAS MODIFICAÇÕES.

TENTE IDENTIFICAR A MODIFICAÇÃO FEITA E TENTE ENTENDER COMO ISSO INFLUENCIA NA EXECUÇÃO DO PROGRAMA.

AS MODIFICAÇÕES ESTÃO EM TORNO DO USO DO SUPER() OU NÃO USO DO SUPER(). E TAMBÉM SE ELE ESTÁ SENDO USADO ANTES OU DEPOIS DO PRINT() EM CADA UMA DAS CLASSES.

1. Sobre o código abaixo:

```
class Primeiro():
    def __init__(self):
        print("Primeiro...")
        super().__init__()

class Segundo():
    def __init__(self):
        print("Segundo...")
        super().__init__()

class Terceiro(Primeiro,Segundo):
    def __init__(self):
        print("terceiro...")
        super(Terceiro,self).__init__()
```

- a) O programa executou corretamente? Em caso NEGATIVO, qual o erro encontrado?
- O código não apresenta nenhum erro e também não imprime nada na tela.

b) Crie um objeto do tipo da classe Terceiro.

t = Terceiro()

c) Será impresso alguma coisa ao final da execução do programa? Se **SIM**, diga o que será impresso.

Terceiro...

Primeiro...

Segundo...

2. Sobre o código abaixo:

```
class Primeiro():  
    def __init__(self):  
        print("Primeiro...")  
        super().__init__()
```

```
class Segundo():  
    def __init__(self):  
        print("Segundo...")  
        super().__init__()
```

```
class Terceiro(Segundo,Primeiro):  
    def __init__(self):  
        print("terceiro...")  
        super(Terceiro,self).__init__()
```

a) O programa executou corretamente? Em caso **NEGATIVO**, qual o erro encontrado?

- O código executou corretamente.

b) Crie um objeto do tipo da classe Terceiro.

t1 = Terceiro()

c) Será impresso alguma coisa ao final da execução do programa? Se **SIM**, diga o que será impresso.

Sim,

Terceiro ...
Segundo ...
Primeiro...

3. Sobre o código abaixo:

```
class Primeiro():  
    def __init__(self):  
        print("Primeiro...")  
        super().__init__()  
  
class Segundo():  
    def __init__(self):  
        print("Segundo...")  
        super().__init__()  
  
class Terceiro(Segundo,Primeiro):  
    def __init__(self):  
        print("terceiro...")
```

- a) O programa executou corretamente? Em caso NEGATIVO, qual o erro encontrado?
- O código não apresentou nenhum erro.
- b) Crie um objeto do tipo da classe Terceiro.
t2 = Terceiro()
- c) Será impresso alguma coisa ao final da execução do programa? Se **SIM**, diga o que será impresso.
- Sim,
Terceiro...

4. Sobre o código abaixo:

```
class Primeiro():  
    def __init__(self):  
        print("Primeiro...")  
  
class Segundo():  
    def __init__(self):
```

```
print("Segundo...")
```

```
class Terceiro(Segundo,Primeiro):  
    def __init__(self):  
        print("terceiro...")  
        super(Terceiro,self).__init__()
```

- a) O programa executou corretamente? Em caso NEGATIVO, qual o erro encontrado?
 - O programa executou corretamente.
- b) Crie um objeto do tipo da classe Terceiro.
t4 = Terceiro()
- c) Será impresso alguma coisa ao final da execução do programa? Se **SIM** , diga o que será impresso.
Terceiro...
Segundo...

5. Sobre o código abaixo:

```
class Primeiro():  
    def __init__(self):  
        print("Primeiro...")  
  
class Segundo():  
    def __init__(self):  
        print("Segundo...")  
        super().__init__()  
  
class Terceiro(Segundo, Primeiro):  
    def __init__(self):  
        print("terceiro...")  
        super(Terceiro,self).__init__()
```

- a) O programa executou corretamente? Em caso NEGATIVO, qual o erro encontrado?
 - O programa executou corretamente.
- b) Crie um objeto do tipo da classe Terceiro.
t5 = Terceiro()

- c) Será impresso alguma coisa ao final da execução do programa? Se **SIM** , diga o que será impresso.

Terceiro ...

Segundo ...

Primeiro ...

6. Sobre o código abaixo:

```
class Primeiro():
    def __init__(self):
        super().__init__()
        print("Primeiro...")

class Segundo():
    def __init__(self):
        super().__init__()
        print("Segundo...")

class Terceiro(Segundo, Primeiro):
    def __init__(self):
        super(Terceiro,self).__init__()
        print("terceiro...")
```

- d) O programa executou corretamente? Em caso **NEGATIVO**, qual o erro encontrado?

- O programa executou corretamente.

- e) Crie um objeto do tipo da classe Terceiro.

t6 = Terceiro()

- f) Será impresso alguma coisa ao final da execução do programa? Se **SIM** , diga o que será impresso.

- Sim

Primeiro ...

Segundo ...

Terceiro ...

7. Qual a função do `super().__init__()` das classes `Primeiro()` e `Segundo()`?

- Usando o `super().__init__` , as classes `Primeiro()` e `Segundo`, irão herdar o `init` da classe `Object()`

8. Qual a função do `super(Terceiro,self).__init__()` da classe `Terceiro()`?

- O `super(Terceiro,self).__init__()` da classe `Terceiro()` irá fazer com que a classe `Terceiro()` herde o `init` das suas superclasses.