

EXERCÍCIO SOBRE LISTAS E IF/ELSE

1) Considerando a lista inicializada logo abaixo, escreva um comando para resolver cada item solicitado. (Obs: utilize apenas os conteúdos vistos nas aulas)

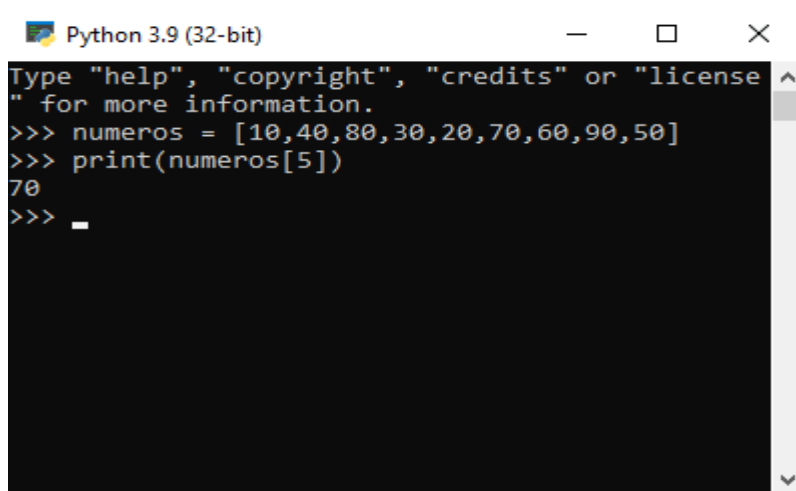
`numeros = [10,40,80,30,20,70,60,90,50]`

a) Apresente na tela o elemento do índice 5.

R.: `numeros = [10,40,80,30,20,70,60,90,50]`

`print(numeros[5])`

`70`



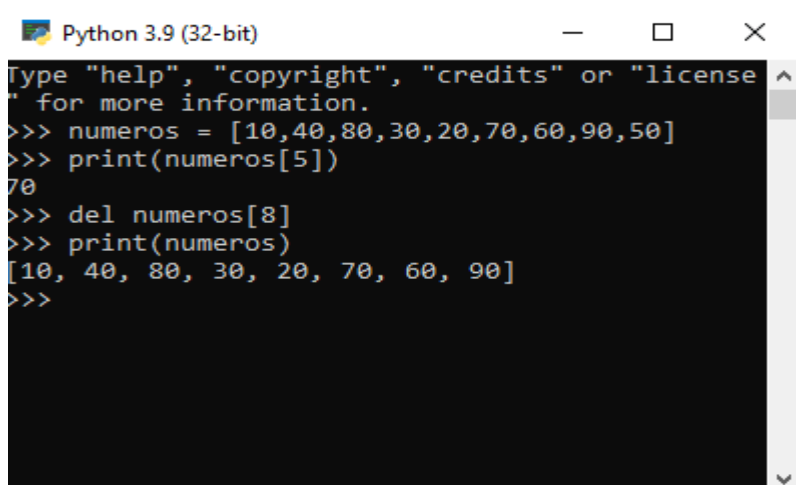
```
Python 3.9 (32-bit)
Type "help", "copyright", "credits" or "license()" for more information.
>>> numeros = [10,40,80,30,20,70,60,90,50]
>>> print(numeros[5])
70
>>> _
```

b) Delete o último elemento da lista.

R.: `del numeros[8]`

`print(numeros)`

`[10,40,80,30,20,70,60,90]`



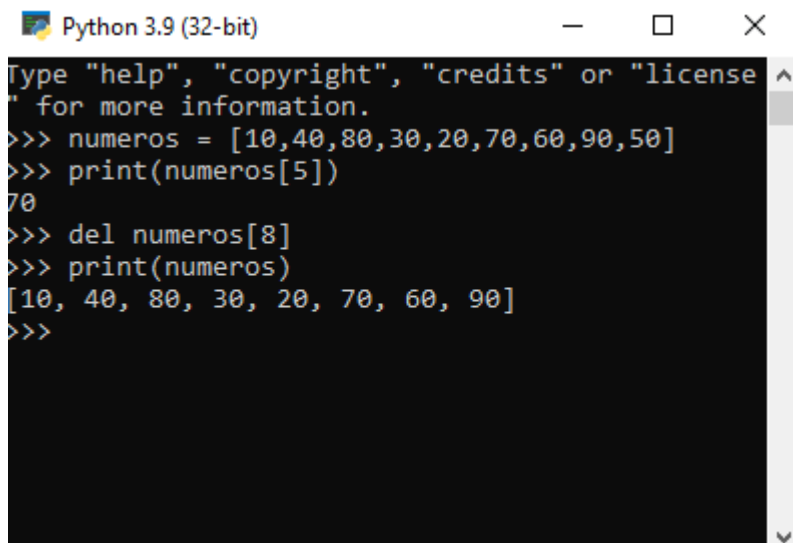
```
Python 3.9 (32-bit)
Type "help", "copyright", "credits" or "license()" for more information.
>>> numeros = [10,40,80,30,20,70,60,90,50]
>>> print(numeros[5])
70
>>> del numeros[8]
>>> print(numeros)
[10, 40, 80, 30, 20, 70, 60, 90]
>>>
```

c) Adicione o elemento 100 no índice 2 da lista.

R.: `numeros.insert(2,100)`

`print(numeros)`

`[10,40,100,80,30,20,70,60,90]`

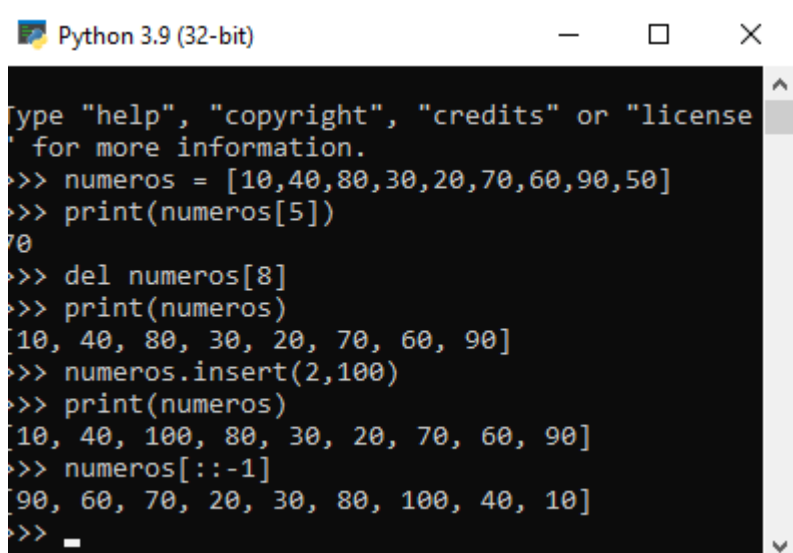


```
Python 3.9 (32-bit)
Type "help", "copyright", "credits" or "license()" for more information.
>>> numeros = [10,40,80,30,20,70,60,90,50]
>>> print(numeros[5])
70
>>> del numeros[8]
>>> print(numeros)
[10, 40, 80, 30, 20, 70, 60, 90]
>>>
```

d) Mostre os elementos na ordem contrária (direita para esquerda).

R.: `numeros[::-1]`

`[90,60,70,20,30,80,100,40,10]`

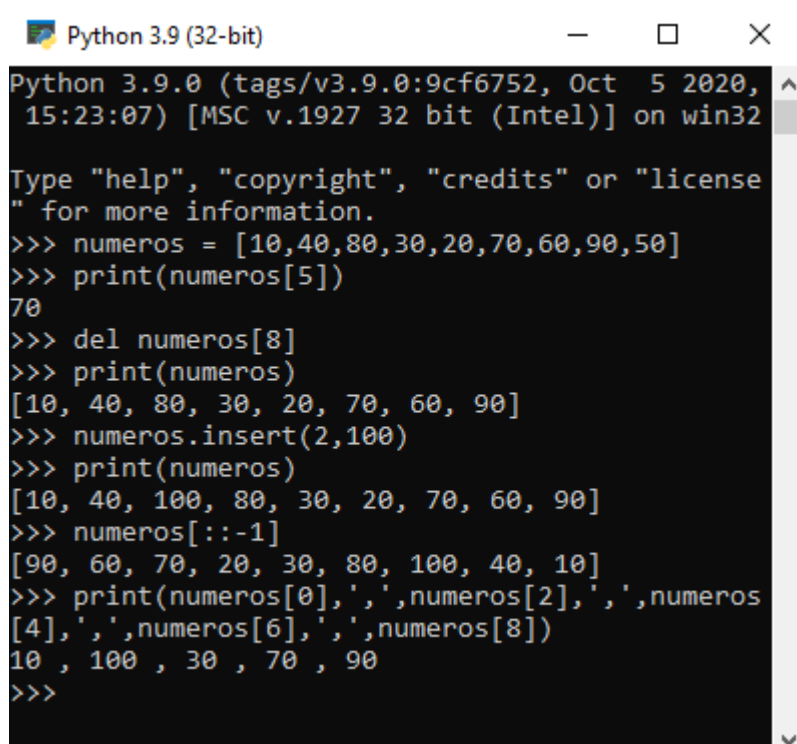


```
Python 3.9 (32-bit)
Type "help", "copyright", "credits" or "license()" for more information.
>>> numeros = [10,40,80,30,20,70,60,90,50]
>>> print(numeros[5])
70
>>> del numeros[8]
>>> print(numeros)
[10, 40, 80, 30, 20, 70, 60, 90]
>>> numeros.insert(2,100)
>>> print(numeros)
[10, 40, 100, 80, 30, 20, 70, 60, 90]
>>> numeros[::-1]
[90, 60, 70, 20, 30, 80, 100, 40, 10]
>>>
```

e) Apresente apenas os elementos que estão em índices pares.

R.: `print(numeros[0],',',numeros[2],',',numeros[4],',',numeros[6],',',numeros[8])`

`[10,100,30,70,90]`



```
Python 3.9 (32-bit)
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:23:07) [MSC v.1927 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> numeros = [10,40,80,30,20,70,60,90,50]
>>> print(numeros[5])
70
>>> del numeros[8]
>>> print(numeros)
[10, 40, 80, 30, 20, 70, 60, 90]
>>> numeros.insert(2,100)
>>> print(numeros)
[10, 40, 100, 80, 30, 20, 70, 60, 90]
>>> numeros[::-1]
[90, 60, 70, 20, 30, 80, 100, 40, 10]
>>> print(numeros[0],',',numeros[2],',',numeros[4],',',numeros[6],',',numeros[8])
10 , 100 , 30 , 70 , 90
>>>
```

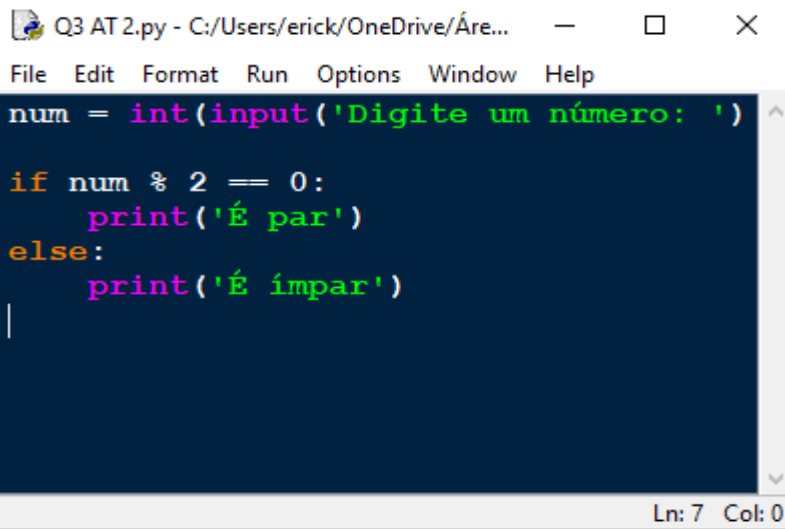
2) Assista ao vídeo (anexo da postagem) sobre estruturas IF-ELSE e explique como elas funcionam em Python.

R.: As condições IF(SE) e ELSE(SENÃO) servem para serem tomadas decisões em uma determinada situação. Se o IF, primeira condição for verdadeira, a condição ELSE não será necessária para ser verificada já que é falsa. Então, tudo que estiver dentro do IF será executado. Caso a condição IF seja falsa o ELSE será executado.

3) Escreva um programa em Python que recebe um número do teclado e através de uma estrutura IF-ELSE informa se ele é ímpar ou par.

R.: `num = int(input('Digite um número: '))`

```
if num % 2 == 0:
    print('É par')
else:
    print('É ímpar')
```



The image shows a screenshot of a Python IDE window. The title bar at the top reads "Q3 AT 2.py - C:/Users/erick/OneDrive/Áre...". Below the title bar is a menu bar with the following options: File, Edit, Format, Run, Options, Window, and Help. The main area of the window is a dark blue editor with Python code written in a light green font. The code is as follows:

```
num = int(input('Digite um número: '))  
  
if num % 2 == 0:  
    print('É par')  
else:  
    print('É ímpar')
```

At the bottom right of the editor, the status bar indicates "Ln: 7 Col: 0".