



30 September 2019

# Introdução ao Python

Controle de Fluxo

SIDIA - Setembro/2019

**Jailson P. Januário**

[jpj.ads@uea.edu.br](mailto:jpj.ads@uea.edu.br)

Github: @jailsonpj



# Desvio Condicional

**if** condição:

comandos

**elif** condição:

comandos

**else:**

comandos

```
1 a = int(input('Digite um número: '))
2
3 #Bloco de desvio condicional
4 if a % 2 == 0:
5     print('Par')
6 else:
7     print('Ímpar')
8
9
10
```

Não precisa de (parêntesis), mas podem ser usados



## Desvio Condicional

- Não existe SWITCH/CASE, quando é necessário várias comparações, usamos **elif's** ou **dicionários** (explicado mais à frente)
- Pode ser escrito em um linha:
  - TRUE **if** condição **else** FALSE

```
print('par' if numero%2 == 0 else 'impar')
```



## Loops - while

- O **while** é *quase* igual às outras linguagens:

**while** condição:

comandos

**else:**

comandos

```
10
11
12 spam = 0
13 while spam < 5:
14     print('Alô SIDIA')
15     spam = spam + 1
16
17
```

- O **else** só é executado se não ocorreu nenhum **break** durante o **while**, ou seja, se o **while** parar pela condição.



## Loops - break e continue

- **break** é um atalho para fazer a execução de um programa sair previamente de uma cláusula de loop *while*.
- Quando a instrução **continue** é executada no programa, ela retorna imediatamente ao início do loop e a condição será avaliada novamente.

```
21
22 while True:
23     print('Digite um nome.')
24     name = input()
25     if name == 'jailson':
26         break
27     print('Obrigado')
28
29
```

```
30
31 while True:
32     print('Qual é seu nome?')
33     name = input()
34     if name != 'Joe':
35         continue
36     print('Alô, Joe. Qual sua senha?')
37     key = input()
38     if key != 'swordfish':
39         break
40     print('Acesso concedido')
41
```



## Loops - for

- O **for** em python é parecido com o **foreach** de outras linguagens:

**for** variavel **in** sequencia:

comandos

**else:**

comandos

```
44  
45 for letra in 'string':  
46     print(letra)  
47  
48 #O |  
49  
50 for i in range(5):  
51     print(i)
```



## Loops - argumentos de incremento

- a função **range()** retorna uma lista de números e pode ser chamada com vários argumentos separados por uma vírgula. Isso permite alterar o inteiro passado de modo que a seguir uma sequência de inteiros, incluindo começar com um número diferente de zero.
- `range(10)`
- `for i in range(10):`
- `for i in range(5,10):`
- `for i in range(0,10,2):`
- `for i in range(5,-1,-1):`



## Encerrando um programa

- Podemos fazer um programa ser encerrado utilizando a função **sys.exit()**. Como essa função está no módulo **sys**, devemos importar **sys** antes de usar no programa.

```
1  '''
2  Exemplos - Curso SIDIA Setembro/2019
3
4  '''
5
6  import sys
7
8  while True:
9      print('Type exit to exit.')
10
11      response = input()
12
13      if response == 'exit':
14          sys.exit()
15
16      print('You typed ' + response + '.')
17
18
```





## Exercícios

- Leia dois números e imprima o maior deles;
- Leia uma letra, se for 'M' imprima 'Masculino', se for 'F' imprima 'Feminino', senão imprima 'Sexo invalido';
- Some os números impares de 0 a 100;
- Imprima a tabuada de um número digitado pelo usuário.