

Seção 2 - Introdução ao Python

Essa aula é inspirada em:

- <https://docs.python.org/3/tutorial/introduction.html> (<https://docs.python.org/3/tutorial/introduction.html>)
- <https://docs.python.org/3/library/stdtypes.html#iterator-types>
(<https://docs.python.org/3/library/stdtypes.html#iterator-types>)

2.1 Iniciando os estudos

Tipo de variáveis (*built-in types*):

- int = número inteiro
- float = número real
- complex = número complexo
- str = texto

Estes tipos de valores podem ser armazenados dentro de variáveis.

Primeiras funções

Função	Descrição
print()	exibe um dado input
input()	retorna uma string dada pelo usuário
int(), float(), str()	conversão de tipo
type()	retorna o tipo da variável

In [2]:

```
print('Olá mundo')
```

Olá mundo

In [5]:

```
variavel_b = input('Digite um numero ')
```

Digite um numero 5

In [8]:

```
variavel_b
```

Out[8]:

```
'5'
```

In [11]:

```
b = 5.  
  
type(b)
```

Out[11]:

float

2.2 Números

Operações básicas:

Símbolo	Descrição
(+)	soma
(-)	subtração
(*)	multiplicação
(/)	divisão
(//)	parte inteira da divisão
(**)	exponencial
+=	adiciona para a variável
*=	multiplica pela variável

- LEE, Kent D. Python Programming Fundamentals. Second Edition. Springer - Verlag London 2014.
- https://link.springer.com/chapter/10.1007%2F978-1-4471-6642-9_8
(https://link.springer.com/chapter/10.1007%2F978-1-4471-6642-9_8)
- https://link.springer.com/chapter/10.1007%2F978-1-4471-6642-9_9
(https://link.springer.com/chapter/10.1007%2F978-1-4471-6642-9_9)

In [19]:

```
a = 5  
b = 5.  
c = a + b  
type(c)
```

Out[19]:

float

In [26]:

```
a = 5  
d = 10  
d += 5  
d
```

Out[26]:

15

In [27]:

```
e = 10**2  
e
```

Out[27]:

100

In [35]:

```
n_1 = 1j  
n_2 = 1j  
n_1*n_2
```

Out[35]:

(-1+0j)

In [38]:

```
string_numero = '123'  
n_1 = float(string_numero)  
  
n_1*n_1
```

Out[38]:

15129.0

In []:

2.3 String

- Strings são variáveis de texto
- Alguns caracteres, como as aspas, não podem ser adicionados diretamente. Então é necessário \
- \n representa uma nova linha
- Algumas operações básicas funcionam para manipular strings

In [55]:

```
a = 'Olá mundo\n'  
b = a**2  
print(b)
```

```
-----  
TypeError                                Traceback (most recent call last)  
<ipython-input-55-e26e3ebb5554> in <module>()  
      1 a = 'Olá mundo\n'  
----> 2 b = a**2  
      3 print(b)
```

TypeError: unsupported operand type(s) for ** or pow(): 'str' and 'int'

In []:

2.4 Listas

Listas são conjuntos de dados que podem conter dados de diferentes tipos. Elas possuem valores indexados a partir de zero.

In [60]:

```
lista_a = [1, 'asdf', [3., 2], 4]  
lista_a
```

Out[60]:

```
[1, 'asdf', [3.0, 2], 4]
```

In [71]:

```
lista_b = [3, 'asdf', 5, 1, 2]  
  
lista_b[2:]
```

Out[71]:

```
[5, 1, 2]
```

In [64]:

```
lista_c = [1, 2, 3, 4]  
lista_c + lista_b
```

Out[64]:

```
[1, 2, 3, 4, 3, 'asdf', 5, 1, 2]
```

In []:

2.5 Similaridades entre listas e strings

- Ambos são indexáveis com índices a partir de [0]
- Strings, assim como listas, podem ser fatiadas
- A função len(x) retorna o número de elementos de uma lista e o número de caracteres de uma string
- A operação de soma funciona de maneira parecida
- Ambas podem ser multiplicadas por um valor inteiro

Mas atenção, não são do mesmo tipo!

In [92]:

```
Lista_1 = ['PA', 'YA', 'TA', 'HA', 'OA', 'NA']  
string_1 = 'PAYATAHAOANA'
```

```
Lista_1 *3
```

Out[92]:

```
['PA',  
'YA',  
'TA',  
'HA',  
'OA',  
'NA',  
'PA',  
'YA',  
'TA',  
'HA',  
'OA',  
'NA',  
'PA',  
'YA',  
'TA',  
'HA',  
'OA',  
'NA']
```

In []:

In []:

In []:

In []:

2.6 Valores Booleanos (*Boolean Values*)

O que é

As variáveis do tipo *bool* armazenam constantes do tipo *True* ou *False*

In [96]:

```
False
```

Out[96]:

```
False
```

In [98]:

```
a = True  
type(a)
```

Out[98]:

```
bool
```

In []:

Comparações

Comparações retornam variáveis do tipo *bool*

Comparação	Descrição
>	menor que
<=	menor ou igual a
>	maior que
>=	maior ou igual a
==	igual a
!=	diferente de

Fonte: <https://docs.python.org/3/library/stdtypes.html#iterator-types>
(<https://docs.python.org/3/library/stdtypes.html#iterator-types>)

In [103]:

```
a = 5 != 6  
a
```

Out[103]:

```
True
```