

# Python para Ciência de Dados - Aula 02

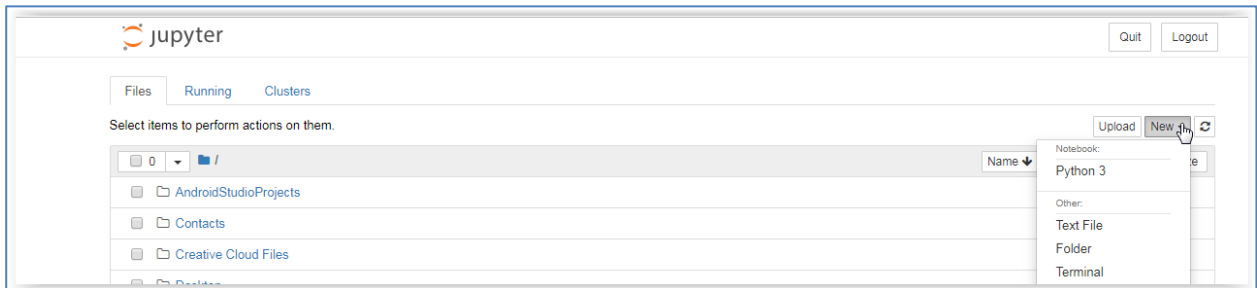
---

## Sumário

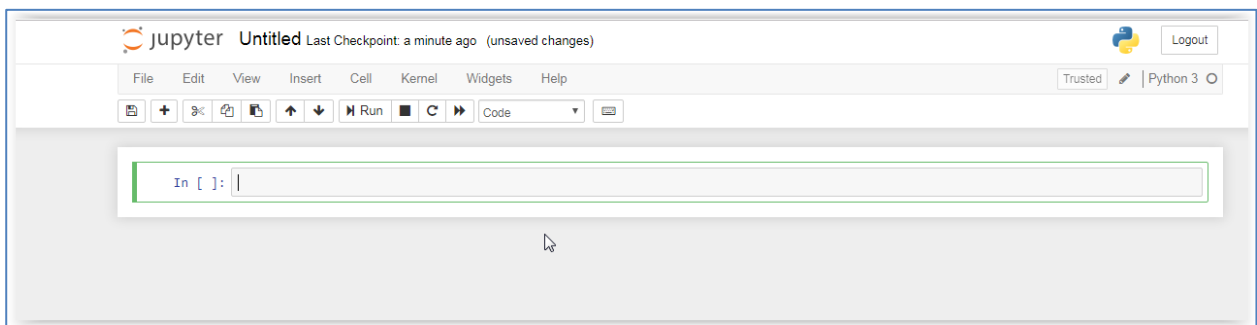
|      |   |   |
|------|---|---|
| 1.   | Executando o Python via Jupyter Notebook..... | 2 |
| 2.   | Variáveis .....                               | 2 |
| 2.1. | Declaração implícita .....                    | 2 |
| 2.2. | Entrada e saída de dados.....                 | 3 |
| 2.3. | Casting.....                                  | 3 |

## 1. Executando o Python via Jupyter Notebook

- Acesse o site <https://www.anaconda.com/>
- Inicie o Jupyter Notebook
- Crie uma pasta para os seus arquivos
- Selecione a referida pasta e selecione New -> Python 3



- No seu navegador padrão deve abrir uma janela assim:



- Para executar basta pressionar CTRL+ENTER

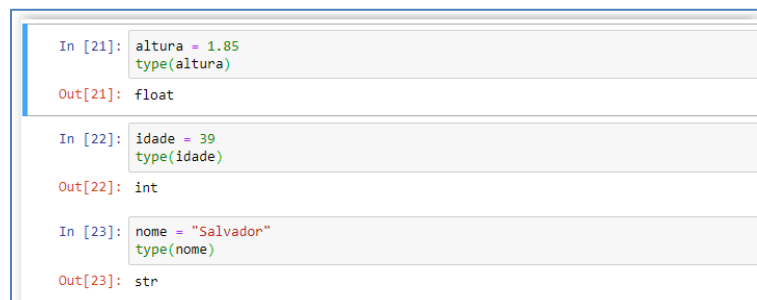
## 2. Variáveis

Como já vimos em outras disciplinas de programação, cujo conceito é o mesmo. Variáveis são posições da memória reservada para armazenar um determinado valor, que tem um nome específico associado a cada variável. No caso específico do Python, o nome da variável deve atender às seguintes regras:

- Começar com letra ou underline;
- Se for um nome composto, não pode conter espaço em branco;
- Os nomes são “case sensitive”;
- Use nomes que sejam auto-explicativos.

### 2.1. Declaração implícita

No python você não precisa definir o tipo de uma variável, que podem ser em geral do tipo: string, integer e float. Basta atribuir um valor a uma variável, depois, se quiser, podemos saber o tipo do dado armazenado usando type.



Uma mesma variável pode receber diferentes tipos de dados, um de cada vez.

```
In [34]: valor = 2.5
         type(valor)
Out[34]: float

In [35]: valor = "teste"
         type(valor)
Out[35]: str
```

## 2.2. Entrada e saída de dados

Podemos utilizar duas funções simples para entrada e saída de dados: `input( )` e `print( )`.

**`variável = input("texto")`**

Onde:

- `variável` = nome da variável que irá receber o valor lido
- `"texto"` é a pergunta que você pode fazer ao usuário

**`print("texto {} texto{}".format(variável1,variável2))`**

Onde:

- `texto` é parte da informação que você quer mostrar na tela
- `{}` é o local do texto onde você quer inserir um valor
- `format()` é uma função que converte o valor para uma representação formatada

A entrada de dados com `input` será sempre lida como um string.

**Exemplo:**

Código

```
idadeGoku = input("Qual a idade do Goku?\n")
print(type(idadeGoku))
```

Saída

```
Qual a idade do Goku?
44
<class 'str'>
```

## 2.3. Casting

É a operação que muda um valor de um tipo para outro. Por exemplo, sabemos que a entrada da função `input` retorna sempre um string. Se quisermos converter de string para inteiro, devemos usar o casting.

**Exemplo:**

Código

```
idadeGoku = input("Qual a idade do Goku?\n")
print(type(idadeGoku))
print(type(int(idadeGoku)))
```

Saída

```
Qual a idade do Goku?
```

```
44
<class 'str'>
<class 'int'>
```

Porque às vezes é necessário mudar o tipo, veja o exemplo abaixo. Se não converter, o resultado pode não ser o que você queria.

### Exemplo:

#### Código

```
valor1 = "4"
valor2 = "5"
print(valor1+valor2)
print(int(valor1)+int(valor2))
```

#### Saída

```
45
9
```