

Computação I - Python

Aula 10: Entrada, Saída e Programa Principal

A função input

Apresentado por: Rafael Machado Andrade

Produção DCC-UFRJ

Metodologia de referência <https://doi.org/10.5753/wei.2016.9683>



A função `input`

Muitas vezes desejamos que o usuário insira alguma informação nos nossos programas durante a sua execução. Essa informação pode ser salva em uma variável e então pode ser reutilizada ao longo do programa.

A função **`input`** é utilizada para esta finalidade: ler informações digitadas pelo usuário.

A função input

Exemplo: Faça uma função que solicite ao usuário que digite seu nome, e salve-o em uma variável.

A função input

Exemplo: Faça uma função que solicite ao usuário que digite seu nome, e salve-o em uma variável.

```
def solicitaNome():  
    '''Funcao que solicita o nome do usuario.  None -> str'''  
    print('Qual o seu nome?')  
    nome = input()  
    return nome
```

```
>>> nome = solicitaNome()  
Qual o seu nome?  
Rafael
```

```
>>> nome  
'Rafael'
```

Observe que a função não recebe nenhum argumento, já que o valor do nome será inserido pelo usuário durante a execução da função. Como o nome inserido é retornado, o retorno da função pode ser atribuído a uma variável, para ser reutilizado.

A função input

Podemos reescrever a função solicitaNome(), embutindo a mensagem contida no comando print dentro do próprio comando de input:

```
def solicitaNome():  
    '''Funcao que solicita o nome do usuario.  None -> str'''  
    nome = input('Qual o seu nome?')  
    return nome
```

```
>>> nome = solicitaNome()  
Qual o seu nome?Rafael
```

```
>>> nome  
'Rafael'
```

A função input

Podemos reescrever a função solicitaNome(), embutindo a mensagem contida no comando print dentro do próprio comando de input:

```
def solicitaNome():  
    '''Funcao que solicita o nome do usuario.  None -> str'''  
    nome = input('Qual o seu nome?')  
    return nome
```

```
>>> nome = solicitaNome()  
Qual o seu nome?Rafael
```

```
>>> nome  
'Rafael'
```

Observe que não houve uma quebra de linha após a mensagem.

Ao se incluir a mensagem diretamente na função input, a quebra de linha ao final da mensagem não é incluída automaticamente como ocorre ao se utilizar a função print.

A função input

Podemos reescrever a função solicitaNome(), embutindo a mensagem contida no comando print dentro do próprio comando de input:

```
def solicitaNome():  
    '''Funcao que solicita o nome do usuario.  None -> str'''  
    nome = input('Qual o seu nome?\n')  
    return nome
```

```
>>> nome = solicitaNome()  
Qual o seu nome?  
Rafael
```

```
>>> nome  
'Rafael'
```

Podemos incluir o caractere '\n' ao final da mensagem para inserir manualmente uma quebra de linha após a ela.

A função input

OBSERVAÇÃO: Como a função input recebe um valor a partir de um texto inserido pelo usuário, seu conteúdo sempre será identificado como sendo de tipo textual (str). Caso queiramos que os dados digitados pelo usuário sejam recebidos como valores numéricos ou booleanos, será preciso realizar a conversão do texto digitado para o tipo de dados desejado.

Essa conversão pode ser feita diretamente ao se executar a função input.

A função input

Exemplo: Faça uma função que solicite ao usuário sua idade (em anos) e seu peso (em Kg) e retorne uma tupla contendo respectivamente estes dois dados.
OBS: A idade é um número inteiro, e o peso é um número decimal.

A função input

Exemplo: Faça uma função que solicite ao usuário sua idade (em anos) e seu peso (em Kg) e retorne uma tupla contendo respectivamente estes dois dados.
OBS: A idade é um número inteiro, e o peso é um número decimal.

```
def solicialdadeEPeso():  
    '''Funcao que solicita a idade e o peso do usuario.  
    None -> int, float'''  
    idade = int(input("Digite sua idade: \n"))  
    peso = float(input("Digite seu peso (em Kg): \n"))  
    return idade, peso
```

```
>>> idade, peso = solicialdadeEPeso()  
Digite sua idade:  
32  
Digite seu peso (em Kg):  
81.6
```

```
>>> idade, type(idade)  
(32, <class 'int'>)  
>>> peso, type(peso)  
(81.6, <class 'float'>)
```

A função input

Caso não tivéssemos utilizado cast para converter os dois inputs, ambas as variáveis teriam dados do tipo string:

```
def solicitadadeEPeso():  
    '''Funcao que solicita a idade e o peso do usuario.  
    None -> str, str'''  
    idade = input("Digite sua idade: \n")  
    peso = input("Digite seu peso (em Kg): \n")  
    return idade, peso
```

```
>>> idade, peso = solicitadadeEPeso()  
Digite sua idade:  
32  
Digite seu peso (em Kg):  
81.6
```

```
>>> idade, type(idade)  
('32', <class 'str'>)  
>>> peso, type(peso)  
('81.6', <class 'str'>)
```

A função input

OBSERVAÇÃO: É necessário ter cuidado na utilização de cast no comando de input. Caso o dado inserido pelo usuário não seja compatível com o tipo de dado do cast, um erro ocorrerá:

```
>>> idade = int(input('Digite sua idade: \n'))
Digite sua idade:
abc
Traceback (most recent call last):
  File "<pyshell#19>", line 1, in <module>
    idade = int(input('Digite sua idade: \n'))
ValueError: invalid literal for int() with base 10: 'abc'
```

Como tentamos converter os dados digitados para o tipo 'int', e como foram digitadas as letras 'abc', que não podem ser convertidas para tipos numéricos, um erro de valor foi ocasionado.

Computação I - Python

Aula 10: Entrada, Saída e Programa Principal

A função input

Apresentado por: Rafael Machado Andrade

Produção DCC-UFRJ

Metodologia de referência <https://doi.org/10.5753/wei.2016.9683>

