

Computação I - Python

Aula 2 - Função

Argumentos Default

Apresentado por: Carolina G. Marcelino

Produção DCC-UFRJ

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



Argumentos Default

Função

Exemplo: Defina uma função que dados dois inteiros x e y , retorna x^y .

Função

Exemplo: Defina uma função que dados dois inteiros x e y , retorna x^y .
Temos a função que eleva um número ao quadrado:

```
1 def quadrado(X):  
2     """Funcao que retorna o quadrado de um numero"""  
3     return X**2
```

Função

Exemplo: Defina uma função que dados dois inteiros x e y , retorna x^y .
Temos a função que eleva um número ao quadrado:

```
1 def quadrado(X):  
2     """Funcao que retorna o quadrado de um numero"""  
3     return X**2
```

Poderíamos facilmente definir a função *potencia*:

Função

Exemplo: Defina uma função que dados dois inteiros x e y , retorna x^y .
Temos a função que eleva um número ao quadrado:

```
1 def quadrado(X):  
2     """Funcao que retorna o quadrado de um numero"""  
3     return X**2
```

Poderíamos facilmente definir a função *potencia*:

```
1 def potencia(X,Y):  
2     """Funcao que dados os inteiros X e Y retorna X elevado a Y"""  
3     return X**Y
```

Função

Exemplo: Defina uma função que dados dois inteiros x e y , retorna x^y .
Temos a função que eleva um número ao quadrado:

```
1 def quadrado(X):  
2     """Funcao que retorna o quadrado de um numero"""  
3     return X**2
```

Poderíamos facilmente definir a função *potencia*:

```
1 def potencia(X,Y):  
2     """Funcao que dados os inteiros X e Y retorna X elevado a Y"""  
3     return X**Y
```

Na verdade, podemos ficar só com esta função:

potencia(x,2)

```
1 def potencia(X,Y):  
2     """ Funcao que dados os inteiros X e Y retorna X elevado a Y """  
3     return X**Y
```

```
1 >>> potencia(3,2)  
2     9  
3  
4 >>> potencia(2,3)  
5     8
```


Função

Podemos definir a função potencia de outra forma:

```
1 def potencia(X,Y=2):  
2     """Funcao que dados os inteiros X e Y retorna X elevado a Y.  
   Caso o valor de Y nao seja passado, o numero X sera elevado ao  
   quadrado."""  
3     return X**Y
```

O que fizemos foi definir um **argumento default**, ou seja, no exemplo, se o usuário não fornecer o segundo parâmetro, a função considera seu valor igual a 2.

```
1 >>> potencia(5)  
2     25  
3  
4 >>> potencia(5,3)  
5     125
```

Função

Argumentos Default: Permitem que valores default sejam utilizados quando nenhum valor é especificado em um certo parâmetro.

Formato

```
def nome-funcao(arg0, ..., argN, argN+1 = default1, ..., argM = defaultM)  
    ...
```

- *arg*₀, ..., *arg*_N: Argumentos sem valores default.
- *arg*_{N+1} = *default*₁, ..., *arg*_M = *default*_M: Argumentos com valores default. Devem ser sempre os últimos argumentos.

Resumo:

- Neste vídeo vimos como pode ser interessante fazer uso de argumentos default.

Autores

- **João C. P. da Silva** ▶ Lattes
- **Carla Delgado** ▶ Lattes
- **Ana Luisa Duboc** ▶ Lattes

Colaboradores

- **Anamaria Martins Moreira** ▶ Lattes
- **Fabio Mascarenhas** ▶ Lattes
- **Leonardo de Oliveira Carvalho** ▶ Lattes
- **Charles Figueiredo de Barros** ▶ Lattes
- **Fabício Firmino de Faria** ▶ Lattes

Computação I - Python

Aula 2 - Função

Argumentos Default

Apresentado por: Carolina G. Marcelino

Produção DCC-UFRJ

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