

Python - Documentos

Em Python, uma docstring é uma string literal que serve como documentação de diferentes objetos Python, como funções, módulos, classes, bem como seus métodos e pacotes. É a primeira linha na definição de todas essas construções e se torna o valor do atributo `__doc__`.

DocString de uma função

```
def addition(x, y):  
    '''This function returns the sum of two numeric arguments'''  
    return x+y  
print ("Docstring of addition function:", addition.__doc__)
```

Ele produzirá a seguinte **saída** -

```
Docstring of addition function: This function returns the sum of two numeric arguments
```

A docstring pode ser escrita entre aspas simples, duplas ou triplas. No entanto, na maioria das vezes você pode querer um texto descritivo como documentação, portanto, é desejável usar aspas triplas.

Todos os módulos e funções integrados possuem a propriedade `__doc__` que retorna sua docstring.

Doutrina do módulo matemático

```
import math  
  
print ("Docstring of math module:", math.__doc__)
```

Ele produzirá a seguinte **saída** -

```
Docstring of math module: This module provides access to the mathematical functions  
defined by the C standard.
```

Documentação de funções integradas

O código a seguir exibe a docstring da função `abs()` e da função `randint()` no módulo aleatório.

```
print ("Docstring of built-in abs() function:", abs.__doc__)  
import random
```



```
print ("Docstring of random.randint() function:",
random.randint.__doc__)
```

Ele produzirá a seguinte saída -

Docstring of built-in abs() function: Return the absolute value of the argument.

Docstring of random.randint() function: Return random integer in range [a, b], including both end points.

Docstring da classe interna

Os documentos das classes integradas são geralmente mais explicativos, portanto, o texto ocupa várias linhas. Abaixo, verificamos a documentação da classe dict integrada

```
print ("Docstring of built-in dict class:", dict.__doc__)
```

Ele produzirá a seguinte **saída** -

Docstring of built-in dict class: dict() -> new empty dictionary
dict(mapping) -> new dictionary initialized from a mapping object's
(key, value) pairs

dict(iterable) -> new dictionary initialized as if via:

d = {}

for k, v in iterable:

d[k] = v

dict(**kwargs) -> new dictionary initialized with the name=value pairs in the keyword argument list

Doutrina da classe Template

A classe Template é definida no módulo string da biblioteca padrão do Python. Sua doutrina é a seguinte -

```
from string import Template

print ("Docstring of Template class:", Template.__doc__)
```

Ele produzirá a seguinte **saída** -

Docstring of Template class: A string class for supporting \$- substitutions.

Docstring no sistema de ajuda

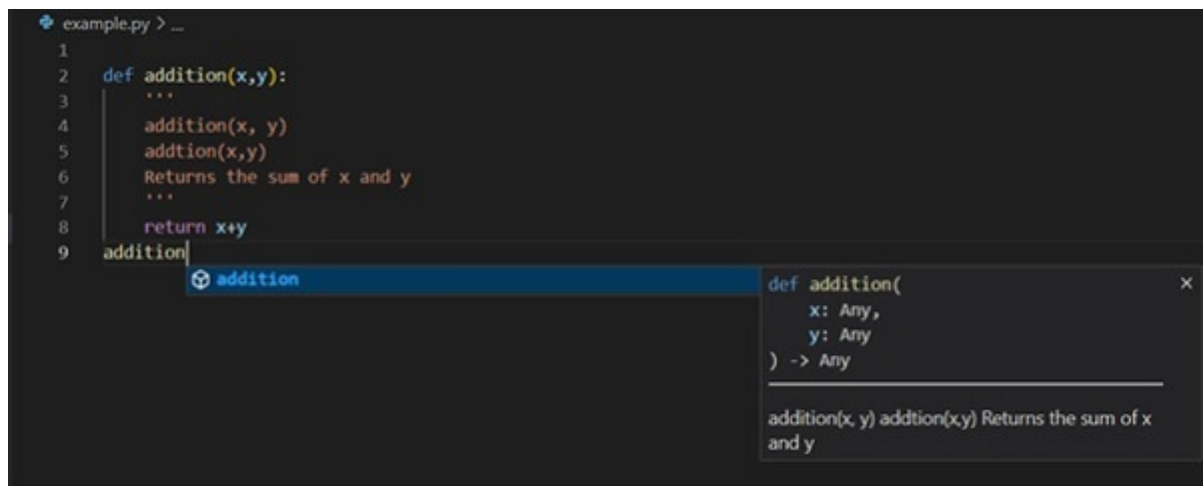
A docstring também é usada pelo serviço de ajuda integrado do Python. Por exemplo, verifique a ajuda da função `abs()` no interpretador Python -

```
>>> help (abs)
Help on built-in function abs in module builtins:
abs(x, /)
    Return the absolute value of the argument.
```

Da mesma forma, defina uma função no terminal interpretador e execute o comando de ajuda.

```
>>> def addition(x,y):
...     '''addtion(x,y)
...     Returns the sum of x and y
...     '''
...     return x+y
...
>>> help (addition)
Help on function addition in module __main__:
addition(x, y)
    addtion(x,y)
    Returns the sum of x and y
```

Docstring também é usado por IDEs para fornecer informações úteis de digitação antecipada durante a edição do código.



Documentação como comentário

Uma string literal que aparece em qualquer lugar que não seja esses objetos (função, método, classe, módulo ou pacote) é ignorada pelo interpretador, portanto, são semelhantes aos comentários (que começam com o símbolo `#`).

```
# This is a comment
print ("Hello World")
'''This is also a comment'''
print ("How are you?")
```