

PYTHON - PART 2 - ADDITIONAL



Autor: Vicent Bosch Vicent.bosch@ceedcv.es 2020/202

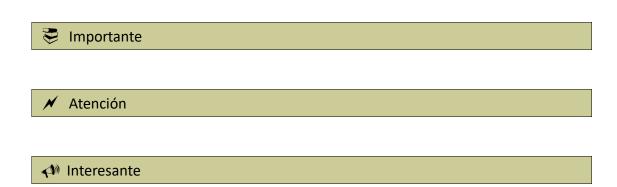
Versión:201126.1112

Licencia

Reconocimiento - NoComercial - Compartirlgual (by-nc-sa): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:



1.	PAS	SS OF ARGUMENTS	4
2.	EX	AMPLES	4
4	2.1	Example 1	4
2	2.2	Example 2	4
2	2.3	Example 3	5

PYTHON - PART 5

1. PASS OF ARGUMENTS

In this unit, you will learn how to execute a Python program from console and how to pass values to the program.

To do this, you need to import the sys library.

The **sys.argv** array stores the name of the program in the first position (0), and the values passed in the other positions.

If you execute from console:

```
py example.py 2 3 4
```

The sys.argv array will contain:

```
sys.argv[0] \rightarrow example.py
sys.argv[1] \rightarrow 2
sys.argv[2] \rightarrow 3
sys.argv[3] \rightarrow 4
```

For more information: https://docs.python.org/3/library/sys.html?highlight=sys%20argv#sys.argv

2. EXAMPLES

2.1 Example 1

```
import sys
print("This is the name of the program:", sys.argv[0])
print("Argument List:", str(sys.argv))
```

Execute this code from your IDE (Visual Studio Code recommended) and check the result.

2.2 Example 2

```
import sys
print("This is the name of the program:", sys.argv[0])
print("This is the first value passed", sys.argv[1])
print("This is the second value passed", sys.argv[2])
print("This is the third value passed", sys.argv[3])
print("Argument List:", str(sys.argv))
```

Save this code as example2.py and go to the terminal, where you saved the file (recommended the workspace created with Visual Studio Code).

Execute it using py command → py example2.py 2 3 4

2.3 Example 3

To validate the number of arguments passed, you can use the *len* function.

```
# We import module sys
import sys
# Len function tell us length of an array. "sys.argv" is an array
# with received parameters
if len(sys.argv) != 3:
    print ('2 parameters are required')
else:
    num1=int(sys.argv[1])
    num2=int(sys.argv[2])
    result=num1+num2
    print(result)
```

Save this code as example3.py and go to the terminal, where you saved the file (recommended the workspace created with Visual Studio Code).

Execute it using py command → py example3.py 2 3

Execute it using py command \rightarrow py example 3.py 2

Execute it using py command \rightarrow py example 3.py

Execute it using py command \rightarrow py example 3.py 2 3 4

```
Python > 🕏 example3.py > ...
      import sys
      if len(sys.argv) != 3:
          print ('2 parameters are required')
      else:
       num1=int(sys.argv[1])
         num2=int(sys.argv[2])
       result=num1+num2
       print(result)
 12
13
PROBLEMS
         OUTPUT DEBUG CONSOLE
                                TERMINAL
PS
                                 Python> py .\example3.py 2 3
5
PS
                                 \Python> py .\example3.py 2
2 parameters are required
                                 Python> py .\example3.py
2 parameters are required
PS
                                \Python> py .\example3.py 2 3 4
2 parameters are required
```