

SINTAXA LIMBAJ PYTHON

Alexandru Nicoi

06 Martie 2021

1 Ce este Python?

- Python este un limbaj de programare interpretat care poate fi executat de pe orice tip de sistem ce suporta interpreter de Python. Ca o paralela, C-ul este un limbaj compilat, ceea ce-l face sa fie compilat doar pentru un anumit tip de sistem, avand librarii particulare pentru sisteme. Deci putem spune ca orice cod pe care-l scriem pe un sistem de operare folosind Python, va putea fi rulat de pe orice alt sistem fara modificari.

```
# Exemplu Hello World
print ("Hello World!")
```

2 Particularitati sintaxa

- Deosebirea mare de care o sa va loviti este faptul ca in Python nu se mai declara tipul variabilei. Daca in alte limbaje trebuie specificat tipul variabilei (int, long, float), in python nu este nevoie, scrieti doar "x = ceva". La C / C++ erati invatati cu acolade pentru a defini un scope, precum si cu ";" dupa fiecare instructiune. Ei bine, in python nu exista asa ceva, astfel ca acest limbaj se bazeaza exclusiv pe spatieri (ii spunem aliniere / indentare). De asemenea, fiecare structura de decizie (if, else, elif) are intotdeauna ":" dupa declarare

```
# Exemplu verificare daca un numar este par
# in C:
# int x = 10;
# if (x % 2 == 0)
#     printf ("%d este par", x);
# else printf ("%d este impar", x);

# in python:
x = 10
if x % 2 == 0:
    print ('{} este par'.format(x))
```

```

else:
    print ('{} este impar'.format(x))

# lucru nou: elif <=> else if in C
# if (x < 0)
#   printf("numar negativ");
# else if (x == 0)
#   printf("numarul este egal cu 0");
# else
#   printf("numar pozitiv");

# in python:
if x < 0:
    print ("numar negativ")
elif x == 0:
    print ("numar egal cu 0")
else:
    print ("numar pozitiv")

```

- Structurile repetitive se declara ceva mai diferit fata de C

```

# while (x > 0)
# {
#   printf("%d\n", x);
#   x--;
# }
while x > 0:
    print (x)
    x = x - 1

# for(int i = 0 ; i < n ; i++)
#   printf("%d" , i);

for i in range(0,n)
    print (i)

```

- Citire de la tastatura, afisare in consola.

```

# in C
# int n;
# scanf("%d", &n);
# printf("%d", n);

# in python
x = int(input())
print (x)

```

- Definirea de scope-uri

```
# in C
# OK = true;
# if (OK == true)
# {
#     printf("primul scope");
#     if (OK == true)
#     {
#         printf("al doilea scope");
#         if (OK == true)
#         {
#             printf("al treilea scope");
#         }
#     }
# }

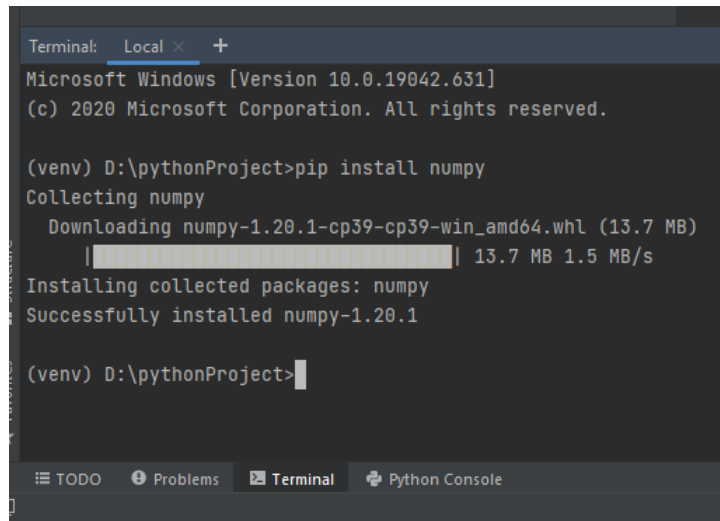
#in python, tineti cont de indentare / spatiere. pentru
#definirea unei noi indentari folositi tasta tab. ca sa
#ajungeti la capat folositi combinatia shift+tab
OK = true
if OK == true:
    print ("primul scope")
    if OK == true:
        print ("al doilea scope")
        if OK == true:
            print ("al treilea scope")
```

3 Import de librarii

- Este de retinut faptul ca Python by-default are putine librarii standard. In antetul programului, se pot importa folosind comanda "import". De asemenea, daca aveti nevoie de o singura instructiune din pachet, puteti da "from pachet import instructiune", sau "import pachet.instructiune". IMPORTANT: Ca sa puteti instala un modul, trebuie sa-l aveti intai instalat in interpreterul de Python.

```
import numpy as np
import matplotlib.pyplot as plt

# pentru a nu folosi in program instructiuni de genul
# numpy.array(), puteti defini alias-uri pentru librarii,
# spre exemplu in cazul lui numpy am folosit aliasul np.
# Pentru pyplot am folosit alias-ul "plt".
# Am inclus mai jos o poza din terminalul din Pycharm cum
# se instaleaza o librerie
# Comanda librerie : "pip install nume-librarie"
```



```
Terminal: Local x +
Microsoft Windows [Version 10.0.19042.631]
(c) 2020 Microsoft Corporation. All rights reserved.

(venv) D:\pythonProject>pip install numpy
Collecting numpy
  Downloading numpy-1.20.1-cp39-cp39-win_amd64.whl (13.7 MB)
    |#####| 13.7 MB 1.5 MB/s
Installing collected packages: numpy
Successfully installed numpy-1.20.1

(venv) D:\pythonProject>
```

4 Librariile Numpy si Pyplot

- Voi la materia Calcul Numeric o sa va folositi mult de librariile numpy si pyplot. Cele mai bune surse de informare despre cum sa le folositi pentru a rezolva problemele primite la aceasta materie le gasiti pe [geekforgeeks](https://www.geeksforgeeks.org/python-numpy/):

<https://www.geeksforgeeks.org/python-numpy/>

<https://www.geeksforgeeks.org/python-introduction-matplotlib/>
