

# Python (#1)

*Nemanja Maček*

- Kraljevski piton (*Python regius*) je među ljubiteljima zmija vrlo zastupljen kućni ljubimac. Neotrovna je zmija poreklom iz Afrike. Zbog svojih dimenzija (konačna dužina tela je između 1 i 1,5 metara), relativno mirnog i staloženog tempermenta odličan je izbor za terariste početnike kao i za “veterane” teraristike. Skromnih je prohteva po pitanju smeštaja i hrane.” [1]



Slika preuzeta sa: <https://www.xyzreptiles.com/product/baby-super-mystic-ball-python/>

- Sintaksa
- Promenljive
- Tipovi promenljivih
- Operatori

## Struktura linije.

```
x=1
if x>0:
    print ("Fizičke i logičke linije.")
```

## Komentari.

```
x=1
# inicijalna vrednost promenljive x je 1 a ovo je prvi komentar.
if x>0:
    print ("Ako dva komentara ovde postoje") # onda je ovo najverovatnije drugi.
```

## Spajanje linija.

```
u=0
```

```
v=1
```

```
w=2
```

```
x=3
```

```
y=4
```

```
z=5
```

```
if u==0 and v>0 \  
    and w>1 and x>2 \  
    and y>3 and z>4:  
    print ("Ovo je primer spajanja linija.")
```

## Više naredbi u liniji.

```
print ("Zdravo Mile")  
print ("Zdravo Lale")  
# ova dva izraza mogu da se napišu na sledeći način  
print ("Zdravo Mile"); print("Zdravo Lale")
```

## Uvlačenje.

- Neispravno:

```
x=1
if x>0:
    print("Ova linija nije uvučena")
print("Komlone, čestitamo, vi ste najbolji programer!")
```



## Uvlačenje.

- Ispravno (blanko karakteri):

```
x=1
if x>0:
    print("Ova linija je uvučena")
    print("Komlene, čestitamo, vi ste najbolji programer!")
```

## Uvlačenje.

- Ispravno (tab):

```
x=1
if x>0:
    print("Ova linija je uvučena")
    print("Komlene, čestitamo, vi ste najbolji programer!")
```

## Uvlačenje.

- Neispravno (nedostaje blanko karakter):

```
if True:  
    print ("Mile")  
    print ("Lale")  
else:  
    print ("Komlen")  
    print ("Programer")
```

- Rezervisane reči:

and	exec	not
assert	finally	or
break	for	pass
class	from	print
continue	global	raise
def	if	return
del	import	try
elif	in	while
else	is	with
except	lambda	yield

## Dodela vrednosti.

```
>>> Item_name = "Computer" # string
>>> Item_qty = 10 # celobrojna vrednost
>>> Item_value = 1000.23 # realni broj
>>> print(Item_name)
Computer
>>> print(Item_qty)
10
>>> print(Item_value)
1000.23
```

## Dodela vrednosti.

- Neispravno:

```
>>> a = 12
```

```
>>> 12 = a
```

```
SyntaxError: can't assign to literal
```

## Višestruko dodeljivanje.

```
>>> x = y = z = 1
```

```
>>> print(x)
```

```
1
```

```
>>> print(y)
```

```
1
```

```
>>> print(z)
```

```
1
```

## Višestruko dodeljivanje.

```
>>> x,y,z = 1,2,"abcd"  
>>> print(x)  
1  
>>> print(y)  
2  
>>> print(z)  
abcd
```



## Promena vrednosti.

```
>>> x = 100
>>> print(x)
100
>>> x = "Python"
>>> print(x)
Python
```

## Zamena (*swap*) vrednosti.

```
>>> x = 10
>>> y = 20
>>> print(x)
10
>>> print(y)
20
>>> x, y = y, x
>>> print(x)
20
>>> print(y)
10
```

## Lokalne i globalne promenljive.

```
var1 = "Python"  
def func1():  
    var1 = "PHP"  
    print("In side func1() var1 = ",var1)  
def func2():  
    print("In side func2() var1 = ",var1)  
func1()  
func2()
```

- Šta dobijamo ako pokrenemo ovaj primer?

## Lokalne i globalne promenljive.

- Ukoliko pokrenete prethodni primer, dobićete:

```
In side func1() var1 = PHP
```

```
In side func2() var1 = Python
```

- Šta se u ovom slučaju dogodilo?

## Lokalne i globalne promenljive.

```
def func1():
    global var1
    var1 = "PHP"
    print("In side func1() var1 = ",var1)
def func2():
    print("In side func2() var1 = ",var1)
func1()
func2()
```

- Šta dobijamo ako pokrenemo ovaj primer?

## Lokalne i globalne promenljive.

- Ukoliko pokrenete prethodni primer, dobićete:

```
In side func1() var1 = PHP
```

```
In side func2() var1 = PHP
```

- Šta se u ovom slučaju dogodilo?

## Brojevi.

```
>>> a=1452
>>> type(a)
<class 'int'>
>>> b=(-4587)
>>> type(b)
<class 'int'>
>>> c=0
>>> type(c)
<class 'int'>
```

## Brojevi.

```
>>> g=1.03
>>> type(g)
<class 'float'>
>>> i=.34
>>> type(i)
<class 'float'>
>>> j=2.12e-10
>>> type(j)
<class 'float'>
>>> k=5E220
>>> type(k)
<class 'float'>
```



## Brojevi.

```
>>> x = complex (1,2)
>>> type(x)
<class 'complex'>
>>> print(x)
(1+2j)
>>> z=1+2j
>>> type(z)
<class 'complex'>
>>> z=1+2J
>>> type(z)
<class 'complex'>
```

## Brojevi.

```
>>> x = True
>>> type (x)
<class 'bool'>
>>> y = false
Traceback (most recent call last):
  File "<pyshell#9>", line 1, in <module>
    y = false
NameError: name 'false' is not defined
>>> y = False
>>> type (y)
<class 'bool'>
>>>
```

## Stringovi.

```
>>> str1 = "String" # double quotes
>>> str2 = 'String' # single quotes
>>> print (str1, str2)
String String
>>> str3 = "String' # ovako ne može
SyntaxError: EOL while scanning string literal
>>> str1 = 'String" # ni ovako ne može
SyntaxError: EOL while scanning string literal
>>> str4 = "Day's" # ovo je OK
>>> str5 = 'Day"s' # ovo je OK
>>> print (str4, str5)
Day's Day"s
```

## Specijalni karakteri u stringovima.

```
>>> print ("This is a backslash (\\) mark.")
This is a backslash (\) mark.
>>> print ("This is a tab \t key.")
This is a tab      key.
>>> print ("These are \'single quotes\'")
These are 'single quotes'
>>> print ("These are \"double quotes\"")
These are "double quotes"
>>> print ("This is a new line \nNew Line")
This is a new line
New Line
```

## Stringovi – indeksi.

```
>>> string1 = "PYTHON TUTORIAL"
```

Karakter	P	Y	T	H	O	N		T	U	T	O	R	I	A	L
Indeks (s leva)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Indeks (s desna)	-15	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

## Stringovi – indeksi.

```
>>> string1 = "PYTHON TUTORIAL"
>>> print (string1[0]) # prikaži prvi karakter
P
>>> print (string1[-15]) # prikaži prvi karakter
P
>>> print (string1[14]) # prikaži poslednji karakter
L
>>> print (string1[-1]) # prikaži poslednji karakter
L
>>> print (string1[4]) # prikaži peti karakter (broji se od nule)
O
>>> print (string1[-11]) # prikaži peti karakter (broji se od -11)
O
```

## Stringovi – indeksi.

```
>>> string1 = "PYTHON TUTORIAL"  
>>> print (string1[16])  
Traceback (most recent call last):  
  File "<pyshell#13>", line 1, in <module>  
    print (string1[16])  
IndexError: string index out of range
```

## Stringovi – seckanje.

```
>>> string1='Python'  
>>> string1[2:4]  
'th'  
>>> string1[-5:]  
'ython'  
>>> string1[:-2]  
'Pyth'
```

Python

0 1 2 3 4 5

-6 -5 -4 -3 -2 -1



## Stringovi – operator “in”.

```
>>> string1 = "PYTHON TUTORIAL"  
>>> 'Z' in string1  
False  
>>> 'P' in string1  
True  
>>> 'TUT' in string1  
True
```

## Stringovi – nepromenljivost.

```
>>> string1 = "PYTHON TUTORIAL"  
>>> string1[0]  
'p'  
>>> string1[0]='A'  
Traceback (most recent call last):  
  File "<pyshell#16>", line 1, in <module>  
    string1[0]='A'  
TypeError: 'str' object does not support item assignment  
>>> string1[0]  
'p'
```

## Tuple.

```
>>> tup1 = (0, -1, 12, 212.23, 100) # pet elemenata
>>> type (tup1)
<class 'tuple'>
>>> print (tup1)
(0, -1, 12, 212.23, 100)
>>> tup2 = ('Red', 'Black', 2000, "White") # četiri elemenata različitih tipova
>>> print (tup2)
('Red', 'Black', 2000, 'White')
>>> tup3 = "a1", "b1", "c1", "d1" # pet elemenata, ne koristimo zagrade
>>> type (tup3)
<class 'tuple'>
>>> print (tup3)
('a1', 'b1', 'c1', 'd1')
```

## Tuple.

```
>>> empty_tup = () # pravimo prazni tuple
>>> print empty_tup
()
>>> single_tup1 = (100,) # pravimo tuple sa jednim elementom
>>> print (single_tup1)
(100,)
```

## Tuple.

```
>>> tup2 = ("Red", "Black", 2000, 12.12)
>>> print (tup2)
('Red', 'Black', 2000, 12.12)
>>> print (tup2[0]) # vrati prvi element
Red
>>> print (tup2[3]) # vrati poslednji element
12.12
>>> print (tup2[4])
Traceback (most recent call last):
  File "<pyshell#47>", line 1, in <module>
    print (tup2[4])
IndexError: tuple index out of range
```

## Tuple – nepromenljivost.

```
>>> tup2 = ("Red", "Black", 2000, 12.12)
>>> print (tup2[0])
Red
>>> tup2[0] = ("White")
Traceback (most recent call last):
  File "<pyshell#50>", line 1, in <module>
    tup2[0] = ("White")
TypeError: 'tuple' object does not support item assignment
>>> print (tup2[0])
Red
```

## Tuple – seckanje.

```
>>> tup2 = ("Red", "Black", 2000, 12.12)
>>> print(tup2[0:2]) # iseckaj prva dva elementa
('Red', 'Black')
>>> print (tup2[1:2]) # secni drugi element
('Black',)
>>> print (tup2[:3]) # secni prva tri elementa
('Red', 'Black', 2000)
>>>
```

## Tuple – operatori + i \*.

```
>>> tup1 = (1,2,3)
>>> tup2 = (4,5,6)
>>> tup3 = (7,8,9)
>>> tup_123 = tup1 + tup2 + tup3 # konkatencija tri tuplea
>>> print (tup_123)
(1, 2, 3, 4, 5, 6, 7, 8, 9)
>>> print (tup1*4) # ponavljanje četiri puta
(1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3)
```



## Liste.

```
>>> my_list1 = [5, 12, 13, 14] # lista koja sadrži celobrojne vrednosti
>>> print (my_list1)
[5, 12, 13, 14]
>>> my_list2 = ['red', 'blue', 'black', 'white'] # lista koja sadrži stringove
>>> print (my_list2)
['red', 'blue', 'black', 'white']
>>> my_list3 = ['red', 12, 112.22] # lista koja sadrži string, int i float
>>> print (my_list3)
['red', 12, 112.22]
>>> my_list=[] # prazna lista
>>> print (my_list)
[]
```

## Liste – indeksi.

```
>>> color_list=["RED", "Blue", "Green", "Black"]
```

Item	RED	Blue	Green	Black
Index (s leva)	0	1	2	3
Index (s desna)	-4	-3	-2	-1

## Liste – indeksi.

```
>>> color_list = ["RED", "Blue", "Green", "Black"]
>>> color_list[0]
'RED'
>>> print (color_list[0],color_list[3])
RED Black
>>> color_list[-1]
'Black'
>>> color_list[4]
Traceback (most recent call last):
  File "<pyshell#94>", line 1, in <module>
    color_list[4]
IndexError: list index out of range
```

## Liste – seckanje.

```
>>> color_list = ["RED", "Blue", "Green", "Black"]
>>> print (color_list[0:2]) # prva dva elementa
['RED', 'Blue']
>>> print (color_list[1:2]) # drugi element
['Blue']
>>> print (color_list[1:-2]) # drugi element
['Blue']
>>> print (color_list[:-3]) # prva tri elementa
['RED']
>>> print (color_list[:]) # kopija originalne liste
['RED', 'Blue', 'Green', 'Black']
```

## Liste – promenljivost.

```
>>> color_list = ["RED", "Blue", "Green", "Black"]
>>> print (color_list[0])
RED
>>> color_list[0]="White" # menjamo vrednost prvog elementa liste
>>> print (color_list)
['White', 'Blue', 'Green', 'Black']
>>> print (color_list[0])
White
```

## Liste – operatori + i \*.

```
>>> color_list1 = ["White", "Yellow"]
>>> color_list2 = ["Red", "Blue"]
>>> color_list3 = ["Green", "Black"]
>>> color_list = color_list1 + color_list2 + color_list3
>>> print (color_list)
['White', 'Yellow', 'Red', 'Blue', 'Green', 'Black']
>>> number = [1,2,3]
>>> print (number[0]*4)
4
>>> print (number*4)
[1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3]
>>>
```

## Skupovi.

```
>>> a = [1, 2, 1, 3, 0, 0, 4, 7, 8, 6]
>>> b = [5, 5, 7, 8, 7, 9, 6, 1, 1, 2]
>>> s1 = set(a) # elementi liste a bez ponavljanja
>>> s2 = set(b) # elementi liste b bez ponavljanja
>>> s1 - s2 # razlika - elementi koji se nalaze u s1 ali ne i u s2
{0, 3, 4}
>>> s1 | s2 # unija - elementi koji se nalaze ili u s1 ili u s2
{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
>>> s1 & s2 # presek - elementi koji se nalaze i u s1 i u s2
{1, 2, 6, 7, 8}
>>> s1 ^ s2 # elementi koji se nalaze u jednom od skupova, ali ne u oba
{0, 3, 4, 5, 9}
```

## Rečnici.

```
>>> pd = {"class":'V', "section":'A', "roll_no":12}
>>> print (pd["class"])
V
>>> print (pd["section"])
A
>>> print (pd["roll_no"])
12
>>> print (pd)
{'class': 'V', 'section': 'A', 'roll_no': 12}
```



## Aritmetički operatori.

```
>>> 14 + 5 # sabiranje
19
>>> 14 - 5 # oduzimanje
9
>>> 14 * 5 # množenje
70
>>> 14 / 5 # deljenje
2.8
>>> 14 % 5 # ostatak pri deljenju
4
>>> 14 ** 5 # stepenovanje
537824
>>> 14 // 5 # floor division
2
```

## Poređenja.

```
>>> x = 12
```

```
>>> y = 15
```

```
>>> x == y
```

```
False
```

```
>>> x != y
```

```
True
```

```
>>> x > y
```

```
False
```

```
>>> x < y
```

```
True
```

```
>>> x >= y
```

```
False
```

```
>>> x <= y
```

```
True
```

## Logički operatori.

```
>>> x = 12
>>> y = 15
>>> (x > 10 and y > 10) # vraća True ako su oba veća od 10
True
>>> (x > 14 or y > 14) # vraća True ako je bar jedan veći od 10
True
>>> not (x > 10 and y > 10) # negacija (vraća true ako je x<=10 i y<=10)
False
>>>
```

## Dodela vrednosti.

```
>>> x = 12
>>> y = 7
>>> x += y # skraceno od x = x + y
>>> print (x)
19
>>> x -= y # skraceno od x = x - y
>>> x *= y # skraceno od x = x * y
>>> x /= y # skraceno od x = x / y
>>> x **= y # skraceno od x = x ** y
>>> x %= y # skraceno od x = x % y
>>> x //= y # skraceno od x = x // y
```

**Pitanja su dobrodošla.**

1. <http://moj.pet-centar.rs/Teraristika/5404-Kraljevski-piton.html>
2. Milan Bjelica (2016): Programski jezik Python. Dostupno u sekciji “download” na sledećem linku: [http://www.etf.bg.ac.rs/etf\\_files/udzbenici/python.pdf](http://www.etf.bg.ac.rs/etf_files/udzbenici/python.pdf)
3. Charles R. Severance (2016): Python for Everybody – Exploring Data Using Python 3. Dostupno u sekciji “download” na stranici predmata i na sledećem linku: [http://do1.dr-chuck.com/pythonlearn/EN\\_us/pythonlearn.pdf](http://do1.dr-chuck.com/pythonlearn/EN_us/pythonlearn.pdf)
4. A Bite of Python. Dostupno u sekciji “download” na sledećem linku (na engleskom jeziku): <https://python.swaroopch.com/>.
5. <https://www.w3resource.com/python/python-tutorial.php>