

Python Fundamentals - 4

Silviu Ojog

LINK***Academy***

Exercițiul cu raza

- Creați un program care va cere de la utilizator valoarea razei și va afișa aria cercului.

```
Enter radius: 4  
Circle area is 50.24
```

Exercițiul cu paritatea

- Creați un program care va prelua de la utilizator un număr și care la ieșire va scrie mesajul dacă numărul este par sau impar.

```
Enter number?  
3  
Number is even: False
```

```
Enter number?  
2  
Number is even: True
```

Print

Printați urmatorul text:

I'm taking the python exam

Print

Printați urmatorul text:

I'm taking the python exam

R:

```
print('I\'m taking the python exam')
```

Print

Ce se printează:

```
print('I\\'\\n')
```

Print

Ce se printează:

R:

I'

Variabile

Ce se printează:

```
1st_var = 'hello'  
print(1st_var)
```


Variabile

Ce se printează:

R:

SyntaxError: invalid syntax

// Variabilele nu pot începe cu numere

Variabile

Ce se printează:

```
pass = 'hello'  
print(pass)
```

Variabile

Ce se printează:

```
pass = 'hello'  
print(pass)
```

R:

SyntaxError: invalid syntax

// Numele de variabilele nu pot fi cuvinte
cheie

Variabile

Ce se printează:

```
greeting = 'hello'
```

```
print(Greeting)
```

```
Greeting = 'How are you'
```

Variabile

Ce se printează:

R:

NameError: name 'Greeting' is not defined

Variabile

Ce se printează:

```
has_greeted = true  
print(has_greeted)
```

Variabile

Ce se printează:

R:

NameError: name 'true' is not defined

Variabile

Ce se printează:

```
print (9//2)
```


Variabile

Ce se printează:

```
print (9//2)
```

R:

4

// Împărțire fără modulo

Variabile

Ce se printează:

```
print (9*"2")
```

Variabile

Ce se printează:

```
print (9*"2")
```

R:

222222222

// Se printează stringul 2 de 9 ori

Variabile

Ce se printează:

```
print ("2"*9.0)
```

Variabile

Ce se printează:

```
print ("2"*9.0)
```

R:

TypeError: can't multiply sequence by non-int of type 'float'

Variabile

Ce se printează:

```
print ("2"*"9")
```

Variabile

Ce se printează:

```
print ("2"*"9")
```

R:

TypeError: can't multiply sequence by non-int of type 'str'

Variabile

Ce se printează:

```
print (9+"2")
```


Variabile

Ce se printează:

```
print (9*"2")
```

R:

TypeError: unsupported operand type(s)
for +: 'int' and 'str'

Variabile

Ce se printează:

```
print (9**2)
```

Variabile

Ce se printează:

```
print (9**2)
```

R:

81

// 9 la puterea 2

Variabile

Ce se printează:

```
print (2*1**2)
```

Variabile

Ce se printează:

```
print (2*1**2)
```

R:

2

// Expresia puterii are prioritate

$2 * (1 ^ 2)$

Variabile

Ce se printează:

```
print ('a'+ 'b'*2+4*'a' )
```

Variabile

Ce se printează:

```
print ('a'+'b'*2+4*'a' )
```

R:

abbaaaa

// Operatorul * are precedență față de
operatorul +

Operatorii

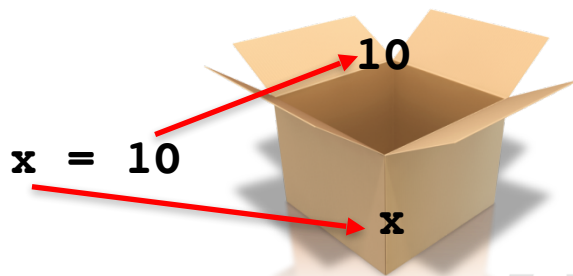
- Operatorii sunt marcaje prin care se execută operația logică sau aritmetică în cadrul codului.
- Operatorii se execută asupra unor valori care în această operație se numesc operanzi.
- Operatorii pot fi binari și ei pot funcționa cu doi operanzi sau unari, care pot funcționa cu un singur operand.
- Partea de cod în care sunt utilizați operatorii se numește expresie.
 - + - * / % and or == != < > <= >=

Operatorii

- Operatorii se clasifică în trei tipuri:
 - Operatori aritmetici
 - Operatori de comparație
 - Operatori logici
- Nu vom greși nici dacă spunem că există încă două tipuri de operatori:
 - Operatori de atribuire
 - Operatori de tip bit

Operatorul de atribuire

- Operandul din partea dreaptă, se atribuie la o anumită locație (operandul din partea stângă).



Operatorul de atribuire

Care linie de cod nu este corectă?

bool = false # line 1

if bool: # line 2

 print ('Python Programming!') # line 3

else: # line 4

 print ('JavaScript Programming!') # line 5

Operatorul de atribuire

Care linie de cod nu este corectă?

```
bool = False # line 1
if bool: # line 2
    print ('Python Programming!') # line 3
else: # line 4
    print ('JavaScript Programming!') # line 5
```

Silviu Ojog

Operatori aritmetici

- Operatorii aritmetici sunt operatori pe care îi folosim pentru realizarea operațiilor de calcul:
- Comportarea acestor operatori corespunde comportării operațiilor matematice cu aceeași denumire.

| | |
|---|-----------|
| + | Adunare |
| - | Scădere |
| * | Înmulțire |
| / | Împărțire |

Silviu Ojog

Operatori aritmetici

- Operatorii aritmetici sunt operatori pe care îi folosim pentru realizarea operațiilor de calcul:
- Comportarea acestor operatori corespunde comportării operațiilor matematice cu aceeași denumire.

| | |
|---|-----------|
| + | Adunare |
| - | Scădere |
| * | Înmulțire |
| / | Împărțire |

Silviu Ojog

Operatori aritmetici speciali

- Pe lângă operatorii aritmetici standard, Python recunoaște și operatori cu scopuri speciale:

| | |
|----|--------------------------------------|
| // | Împărțirea fără rest |
| ** | Puterea |
| % | Restul întreg din împărțire (modulo) |

$10 // 3 = \mathbf{3}$

$10 \% 3 = \mathbf{1}$


$2 ** 3 = \mathbf{8}$

Silviu Ojog

Operatori unari

- Operatorii unari se execută asupra unui operatori și se folosesc pentru schimbarea semnului.

$-X$



x devine negativ

$+X$



x devine pozitiv

Prescurtările

- În programare, deseori modificăm variabilele prin intervenirea asupra valorilor lor. De exemplu

$x = x + 10$

- Altfel, aceasta se poate scrie și astfel:

$x + = 10$

Prescurtările

- Același concept este aplicabil și la toți ceilalți operatorii aritmetici:

`x += 10`

`x = x + 10`

`x -= 10`

`x = x - 10`

`x *= 10`

`x = x * 10`

`x /= 10`

`x = x / 10`

`x %= 10`

`x = x % 10`

`x //= 10`

`x = x // 10`