

Python – 4ο ΓΕΛ Κέρκυρας

Τύποι Δεδομένων

- Αριθμητικοί
 - Ακέραιοι (integer)
0,4,5,6,9 , -40
 - Πραγματικοί (Float)
6.3 , 5.0, 9.3, -11.6
- Χαρακτήρες ή Συμβολοσειρές (String)
'456' , '#abcd5678' , "George" , "Σήμερα είναι Παρασκευή"
- Λογικοί (Boolean)
True / false
- Λίστες
{2,3.14, "b"}, {3,4,[1,2]}
- Σύνολα
{2,3,4 "Γιώργος" , 3.14}

Εντολές

- Εντολές Εισόδου – Εξόδου
 - Input
 - Print
- Εντολή Εκχώρησης
 - num=15
- Εντολές Ελέγχου
 - If... else:
- Εντολές Επανάληψης
 - for , while

Τελεστές

- Αριθμητικοί Τελεστές
+, -, *, /, //, %, **
- Συγκριτικοί Τελεστές
==, <, >, <=, >=, !=
- Λογικοί Τελεστές
and or not

Αριθμητικοί τελεστές

Τελεστής	Όνομα	Εξήγηση	Παράδειγμα
+	Συν	Πρόσθεση αριθμών	6 + 3 δίνει 9 Το 'c' + 'b' δίνει 'cb'
-	Μείον	Αφαίρεση αριθμών	Το 60 - 32 δίνει 28
*			Το 3 * 4 δίνει 12 Το 'fa' * 2 δίνει 'fafa'
/	Διά	Διαίρεση δύο αριθμών. Το αποτέλεσμα είναι πάντα δεκαδικός αριθμός.	Το 6 / 9 δίνει 0.6666666666666666 Το 10 / 2 δίνει 5.0 Το 1 / 2 δίνει 0.5
//	Ακέραια διαίρεση	Διαίρεση δύο αριθμών στρογγυλοποιημένη προς τα κάτω (floor division)	Το 5 // 6 δίνει 0. Το 17 // 3 δίνει 5. Το 17.2 // 3 δίνει 5.0
%	Υπόλοιπο	Υπόλοιπο διαίρεσης δύο αριθμών	Το 13 % 4 δίνει 1
**	Δύναμη	Ύψωση αριθμού σε δύναμη	Το 2 ** 3 δίνει 8 Το 3.0 ** 4 δίνει 81.0 Το 10 ** 3 δίνει 1000

Λογικοί Τελεστές

Λογική Πρόταση Α	Λογική Πρόταση Β	A or B	A and B	not A
True	True	True	True	False
True	False	True	False	False
False	True	True	False	True
False	False	False	False	True

Π.χ.

```
>>> a=5
>>> b=6
>>> a>0 and b<9
True
```

```
>>> a<0 and a!=b
False
>>> a>0 and a!=b
True
>>> not(a>0) or b<a
False
```

Παραδείγματα εντολών

```
print (1)
print ('asdf')
primeNumbers = 1 , 2 , 3 , 5 , 7
print (primeNumbers)
```

Παράδειγμα Ανάθεση αριθμητικών τιμών σε μεταβλητές

Ανάθεση	Επεξήγηση
<code>x = 12</code>	Θέτει <code>x=12</code>
<code>x, y = 3, 5</code>	Θέτει <code>x=3</code> και <code>y=5</code>
<code>x = y = 16.3</code>	Θέτει <code>x=16.3</code> και <code>y=16.3</code>
<code>x = y = 2, 6</code>	Θέτει <code>x=(2,6)</code> και <code>y=(2,6)</code>
<code>x = y = 3,2,6</code>	Θέτει <code>x = (3,2,6)</code> και <code>y = (3,2,6)</code>
Κώδικας	Αποτέλεσμα εκτέλεσης
<pre>x=12 print(x) x, y=3, 5 print(x, y) x=y=16.3 print(x, y) x=y=2, 6 print(x, y) x=y=3, 2, 6 print(x, y)</pre>	<pre>12 3 5 16.3 16.3 (2, 6) (2, 6) (3, 2, 6) (3, 2, 6)</pre>

<pre>print("Χρησιμοποιούμε διπλά εισαγωγικά") print('ή μονά εισαγωγικά') print("Αν περιέχει 'μονά', το βάζω σε διπλά") print('και αντίστροφα "διπλά" μέσα σε μονά') print("Μπορώ και με backslash, π.χ. \' \'"") print("""τριπλά εισαγωγικά για παραπάνω από μια γραμμή\n το \n είναι νέα γραμμή""")</pre>	<pre>Χρησιμοποιούμε διπλά εισαγωγικά ή μονά εισαγωγικά Αν περιέχει 'μονά', το βάζω σε διπλά και αντίστροφα "διπλά" μέσα σε μονά Μπορώ και με backslash, π.χ. ' ' τριπλά εισαγωγικά για παραπάνω από μια γραμμή το \n είναι νέα γραμμή</pre>
--	---

Επιπλέον παράδειγμα χρήσης της print()

<pre>name = "Peter" print("Hi", name, "how are you?") print("Hi"+name+"how are you?")</pre>	<pre>Hi Peter how are you? HiPeterhow are you? >>></pre>
---	---

Χρήση της Python ως Αριθμομηχανής

Μπορούμε να φανταστούμε τη γλώσσα ως μια αριθμομηχανή όπου οι τέσσερις βασικές πράξεις λειτουργούν με τα σύμβολα +, -, *, /. Κάθε φορά που θέλουμε να τυπώσουμε το αποτέλεσμα μιας πράξης γράφουμε την εντολή print. Για παράδειγμα:

```
>>> print 5+3
```

```
8
```

```
>>> print (7-2)*4
```

```
20
```

Η γλώσσα επιτρέπει τη χρήση παρενθέσεων. Επίσης μπορούμε να τυπώνουμε πολλά αποτελέσματα με την ίδια εντολή, αρκεί να χωρίσουμε με κόμματα:

```
>>> print 7+2, 7-2, 7*2, 7/2
```

```
9 5 14 3
```

Μια μικρή προσοχή χρειάζεται με την πράξη της διαίρεσης. Η Python εκτελεί ακέραια διαίρεση, εκτός αν δηλώσουμε ότι θέλουμε να κάνει διαίρεση πραγματικών αριθμών. Ο πιο απλός τρόπος να το δηλώσουμε αυτό είναι να γράψουμε τον διαιρετέο ή τον διαιρέτη (ή και τους δύο) ως πραγματικούς. Πχ.:

```
>>> print 7/2, 7/2.0, 7.0/2, 7.0/2.0, 7./2, 7/2., 7./2.
```

```
3 3.5 3.5 3.5 3.5 3.5 3.5
```

Και για να συμπληρώσουμε τις βασικές πράξεις, η ύψωση σε δύναμη δηλώνεται ως **. Πχ.:

```
>>> print 9**2, 9**0.5
```

```
81 3.0
```

Μεταβλητές:

Για να φυλάξουμε ενδιάμεσα αποτελέσματα, οι περισσότερες αριθμομηχανές διαθέτουν έναν ή περισσότερους **καταχωρητές μνήμης** (Memory Register = MR). Η **Python** επιτρέπει πρακτικά απεριόριστο πλήθος καταχωρητών. Τους ονομάζει **μεταβλητές** (variables) και ο προγραμματιστής τους δίνει ένα όνομα (με έναν ή περισσότερους χαρακτήρες) και μια τιμή ως εξής:

```
>>> a=5
```

Δηλώθηκε η μεταβλητή a με τιμή 5.

```
>>> metavliti2=34.45
```

Ομοίως, δηλώθηκε η μεταβλητή metavliti2 με τιμή 34.45. Και τώρα μπορούμε να τις χρησιμοποιήσουμε σε εκφράσεις σα να ήταν αριθμοί:

```
>>> print a, metavliti2, metavliti2/a
```

```
>>> 5 34.45 6.89
```

Ασκήσεις – Μάθημα 1ο στην Python

4ο ΓΕΛ Κέρκυρας

Άσκηση 1^η

Φτιάξτε ένα πρόγραμμα που θα εμφανίζει το μήνυμα «Καλωσήρθατε στην python»

Απάντηση:

```
print ("Καλωσήρθατε στην Python")
```

Άσκηση 2^η

Μπορείς να σκεφτείς άλλους 3 τρόπους για να φτιάξεις το προηγούμενο πρόγραμμα; (Tip: Χρησιμοποιούμε άλλα «εισαγωγικά» ή μεταβλητή)

Απάντηση:

- a) `print ('Καλωσήρθατε στην Python')`
- b) `Print ("Καλωσήρθατε στην Python")`
- c) `message='Καλωσήρθατε στην Python'`
`print (message)`

Άσκηση 3^η

Έστω 2 μεταβλητές, η

`name= 'Όλυ'`

και η

`surname= 'μπιακος'`

Φτιάχνουμε μία ακόμα μεταβλητή την

`comprine= name+surname`

πατώντας την εντολή

`print(comprine)`

τι αποτέλεσμα θα εμφανιστεί;

Απάντηση:

Θα εμφανίσει: **Ολυμπιακος**

Άσκηση 4^η

Φτιάξτε ένα πρόγραμμα που θα εμφανίζει το ονοματεπώνυμό σας με χρήση μεταβλητών

Απάντηση:

`name=Γιώργος`

`surname=Κεφ`

`onoma=name+surname`

`print (onoma)`

Άσκηση 5^η

Φτιάξτε ένα πρόγραμμα που θα εμφανίζει το αποτέλεσμα της πράξης $752 \cdot 325$

```
print 752*325
```