ΜΕΤΑΒΛΗΤΕΣ

Όπως αναφέρθηκε στην προηγούμενη Δραστηριότητα οι Μεταβλητές είναι **θέσεις Μνήμης** στις οποίες μπορούμε να αποθηκεύουμε δεδομένα πχ αριθμούς ή λέξεις για να τα επεξεργαστούμε στην ροή του προγράμματος . Κάθε μεταβλητή έχει ένα **μοναδικό όνομα** για να ξεχωρίζει από τις υπόλοιπες .
Το όνομα ακολουθεί τους συντακτικούς κανόνες της γλώσσας προγραμματισμού .

Ποια ονόματα δέχεται η Python ;

Μπορούν να περιέχουν γράμματα και αριθμούς, αλλά δεν μπορούν να ξεκινούν με αριθμό.

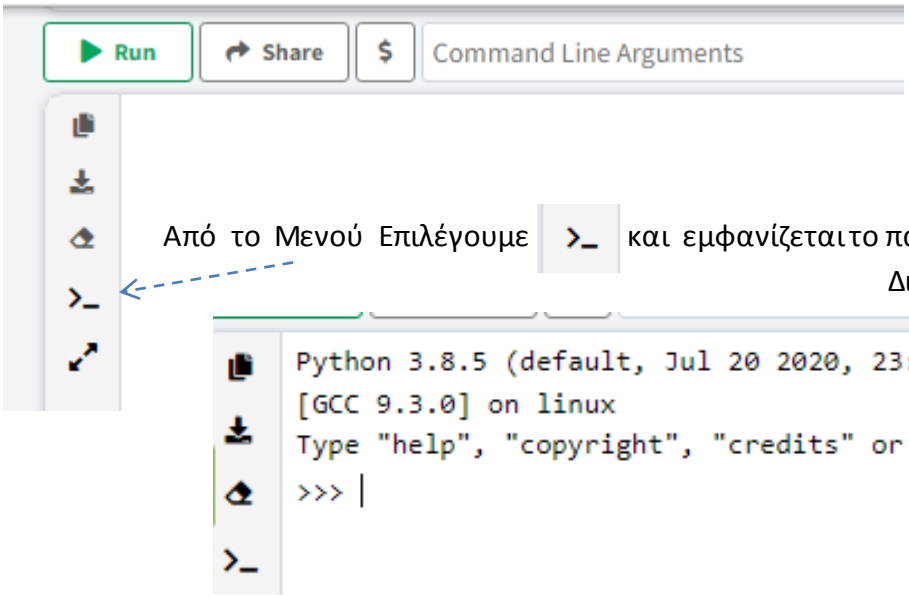
Όταν το όνομα μιας Μεταβλητής αποτελείται από περισσότερες από μια λέξεις , οι λέξεις χωρίζονται με τον χαρακτήρα _ κάτω παύλα.

Παραδείγματα

Σωστές Μεταβλητές : message1 , sum , x , messag1_message2

Λάθος Μεταβλητές : 1message , sum! , 3x , message1-message2

Πως μπορούμε να **ελέγξουμε γρήγορα και εύκολα** τα περιεχόμενα Μεταβλητών ή να δούμε τα αποτελέσματα πράξεων χωρίς να χρησιμοποιήσουμε την εντολή print
Στο πλαίσιο RUN



Εκτελέστε τα παρακάτω παραδείγματα

```
>>> x=5
>>> x
5          ← Εμφανίζεται το περιεχόμενο της μεταβλητής x
>>> 4 + 5
9          ← Εμφανίζεται το αποτέλεσμα 4 + 5
>>> a= 5
>>> b= 10
>>> a + b  ← Εμφανίζεται το περιεχόμενο του αθροίσματος των μεταβλητών a + b
15
>>> c= b - 2
>>> c      ← Εμφανίζεται το περιεχόμενο της μεταβλητής c
8
>>>
```

ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ

Βασικοί Τύποι Δεδομένων

1. Χαρακτήρες (Συμβολοσειρά) (**string**)
2. Ακέραιος **(integer**)
3. Πραγματικός (δεκαδικός) **float**
4. Boolean (Λογικός) Αληθής (**True**) ή Ψευδής (**False**)

Παράδειγμα 1 : **string**

```
>> >message="hello"
>>> message
'hello'
>>>
```

Παράδειγμα 2 : **integer**

```
>>> number=5
>>> number
5
```

Παράδειγμα 3 : **float**

```
>>> number1=1.5
>>> number2=2.8
>>> number1 + number2
4.3
```

Παράδειγμα 4 : **boolean**

```
>>> 1 > 10
      False
>>> 3 < 7
     True
>>>
```

Τι αντιπροσωπεύει το σύμβολο **=** στην Python

Το σύμβολο = σημαίνει αντικατάσταση και όχι ισότητα

Παράδειγμα

X= 5 : Η Μεταβλητή X έχει τιμή 5

X = X + 1 : Η Μεταβλητή X αντικαταστάθηκε με (5+1) δηλ 6

```
>>> x=5
>>> x=x+1
>>> x
6
>>>
```

Τι αντιπροσωπεύει το + όταν έχουμε λέξεις (string)

Ενώνει δύο string

```
>>> message1="Good"
>>> message2="morning"
>>> message1 + message2
'Goodmorning'
```

Σύμβολα αριθμητικών πράξεων :

+ : Πρόσθεση

- : Αφαίρεση

*** : Πολλαπλασιασμός**

/ : Διαίρεση

**** : Δύναμη**

/ / : Ακέραιο μέρος πηλίκου

% : Υπόλοιπο Διαίρεσης

```
>>> 7 + 6
13
>>> 10-2
8
>>> 4 * 3
12
>>> 4 ** 2
 16
>>> 5/2
2.5
 >>> 5//2
2
>>> 8 % 3
2
```

Συγκρίσεις

< : Μικρότερο

<= : Μικρότερο ή ίσον

> : Μεγαλύτερο

>= : Μεγαλύτερο ή ίσον

== : Έλεγχος Ισότητας

!= : Έλεγχος Ανισότητας

```
>>> x=9
>>> x>9
False
>>> x ==9
True
>>> x!=0
True
>>> x >=9
True
>>> x <5
False
```

Λογικοί Τελεστές

not : Λογικό ΟΧΙ

and : Λογικό ΚΑΙ

or : Λογικό Ή

```
>>> x=10
>>> y=20
>>> x>20 and y!=10
False
>>> x!=20 or y>10
True
>>> x==10
True
>>> not x==10
False
```

p	q	not p	p and q	p or q
False	False	True	False	False
False	True	True	False	True
True	False	False	False	True
True	True	False	True	True

ΠΡΟΤΕΡΑΙΟΤΗΤΑ ΠΡΑΞΕΩΝ

Ο ακόλουθος πίνακας μας δίνει τον πίνακα προτεραιοτήτων για την Python, από τη **χαμηλότερη** προς την **υψηλότερη** προτεραιότητα.

Όπου χρειάζεται να αλλάξουμε προτεραιότητα χρησιμοποιούμε **παρενθέσεις**

Είναι πολύ καλύτερο να χρησιμοποιούνται **παρενθέσεις** για να μην γίνονται λάθη όταν δεν γνωρίζουμε ή δεν είναι ξεκάθαρη η προτεραιότητα

ΤΕΛΕΣΤΗΣ	ΠΕΡΙΓΡΑΦΗ
or	
and	**Λογικό ΚΑΙ**
not x	**Λογικό ΟΧΙ**
<, <=, >, >=, !=, ==	**Συγκρίσεις**
+, -	**Πρόσθεση και αφαίρεση**
***, /, //, %**	**Πολλαπλασιασμός, Διαίρεση, Διαίρεση στρογγυλοποιημένη προς τα κάτω, Υπόλοιπο**
****	**Ύψωση σε δύναμη**

ΠΡΟΣΟΧΗ !

Στην έκδοση 3 της python το αποτέλεσμα της διαίρεσης δίνει το ακριβές πηλίκο

```
>>> 5/2
2.5
```

Στην έκδοση 2 το αποτέλεσμα της διαίρεση δίνει το ακέραιο μέρος του πηλίκου

```
>>> 5/2
2
>>>
```

Για να πάρουμε το ακριβές πηλίκο μετατρέπουμε τον διαιρέτη σε πραγματικό (δεκαδικό) αριθμό

```
>>> 5./2
2.5
>>>
```