



Διάλεξη 2^η

Βασικά Αντικείμενα, Δείκτες, Συμβολοσειρές,
Βασική Είσοδος και Έξοδος

Input και Print στην Python

1. `Input()` : Είσοδος δεδομένων από το χρήστη. Διαβάζει μία συμβολοσειρά από το πληκτρολόγιο.
`όνομα = input('Πώς σε λένε? ')`
2. `Print()` : (Εκτύπωση δεδομένων στην οθόνη) ('Γειά σου 'όνομα)
`Print('Καλημέρα')`
`Print(5+3)`

Σημείωση: Η `input` επιστρέφει πάντα τιμές ως συμβολοσειρές (`str`)

Αν θέλουμε αριθμητικές τιμές πρέπει να τις μετατρέψουμε

Π.χ `ηλικία = int(input('Πόσο χρονώ είσαι? '))`

Input και Print στην Python

```
>>> onoma=input('pos se lene? ')
pos se lene? maria
>>> print('geia soy ',onoma)
geia soy  maria
>>>
>>> print('kalimera')
kalimera
>>> print(5+3)
8
>>> 5+3
8
>>>
```

Δείκτες και Τεμάχια

- Στην Python η **θέση** ή ο **δείκτης** ενός χαρακτήρα προσδιορίζεται με ένα από τους αριθμούς 0, 1, 2, 3, ...

s	p	a	m		&		e	g	g	s
↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
0	1	2	3	4	5	6	7	8	9	10

Εικόνα 2.5 Δείκτες των χαρακτήρων στην συμβολοσειρά "spam & eggs".

ΔΕΙΚΤΕΣ

- Εάν `str1` είναι μια μεταβλητή τότε `str1[m:n]` είναι μία υποσυμβολοσειρά που αρχίζει από τη θέση `m` και τελειώνει στη θέση `n-1`.

Παράδειγμα 'kalimera'

```
>>> str1='kalimera'
```

0	1	2	3	4	5	6	7
k	a	l	i	m	e	r	a

```
>>> str1[2:7]
```

```
'limer'
```

Δείκτες-Παραδείγματα

```
>>> str1='Pythoon'  
>>> print(str1)  
Pythoon  
>>> print(str1[0],str1[1],str1[6])  
P y n
```

Η find() μέθοδος: βρίσκει την εγγραφή του ζητούμενου, και επιστρέφει -1 αν δε βρει κάτι.

```
>>>  
>>> str2='Hellow world!'  
>>> print(str2)  
Hellow world!  
>>> print(str2.find('w'))  
5  
>>> print(str2.find('x'))  
-1  
>>> print(str2.find('l'))  
2  
>>> |
```


Αρνητικοί Δείκτες

- Στην Python μπορούμε να χρησιμοποιήσουμε και αρνητικούς δείκτες αρχίζοντας με -1 από δεξιά

-7	-6	-5	-4	-3	-2	-1
p	y	t	h	o	o	n

```
>>> print('python'[-1], 'python'[-4], 'python'[-5:-2])
n h tho
>>>
```

Προεπιλεγμένα Όρια

- Παράδειγμα 3:

```
>>>
>>>
>>> print('Pythoon'[2:], 'Pythoon'[:4], 'Pythoon'[:])
thon Pyth Pythoon
>>> print('Pythoon'[-3:], 'Pythoon'[:-3])
oon Pyth
>>>
```


Συνένωση Συμβολοσειρών

- Δύο συμβολοσειρές μπορούν να ενωθούν και να δημιουργήσουν μια νέα συμβολοσειρά.
 - Αυτό επιτυγχάνεται με το σύμβολο συν (+).
- Ένας συνδυασμός συμβολοσειρών, συμβόλων συν, συναρτήσεων και μεθόδων για τη δημιουργία μιας συμβολοσειράς ονομάζεται **έκφραση συμβολοσειράς**.

Επανάληψη Συμβολοσειρών

- Ο τελεστής αστερίσκος μπορεί να χρησιμοποιηθεί για να συνενώνει επαναληπτικά μια συμβολοσειρά με τον εαυτό της.

Expression	Value
"ha" * 4	"hahahaha"
"mur" * 2	"murmur"
'x' * 10	"xxxxxxxxxx"
("cha-" * 2) + "cha"	"cha-cha-cha"

Χρησιμοποιώντας τα Σχόλια

1. Χρησιμοποιώντας σχόλια άλλοι μπορούν να κατανοήσουν εύκολα το πρόγραμμα. Μια πρόταση σχολίου δημιουργείται με το σύμβολο δέσσης (#), στην αρχή.
2. Μπορείτε να το κατανοήσετε ευκολότερα όταν το διαβάσετε μετά από καιρό
3. Μεγάλα προγράμματα μπορούν να διαβαστούν ευκολότερα, όταν είναι χωρισμένα σε τμήματα.

Μέθοδοι χειρισμού συμβολοσειρών

Σύνταξη: κείμενο.μέθοδος(όρισμα)

str1 = 'Pythoon' a = 'kalimera'

Συνάρτηση	Παράδειγμα	Τιμή	Περιγραφή
len	len(str1)	7	Αριθμός χαρακτήρων
upper	str1.upper()	"PYTHON"	κεφαλαία
lower	str1.lower()	"python"	πεζά
count	str1.count('th')	1	Αριθμός εμφανίσεων
capitalize	a.capitalize()	Kalimera	Το πρώτο γράμμα κεφαλαίο

Παραδείγματα

```
>>> a=('Kalimera')
```

```
>>> a
```

```
'Kalimera'
```

```
>>> a+a
```

```
'KalimeraKalimera'
```

```
>>> a+' '+a
```

```
'Kalimera Kalimera'
```

```
>>> a*3
```

```
'KalimeraKalimeraKalimera'
```

```
>>> a.upper()
```

```
'KALIMERA'
```

```
>>> a.lower()
```

```
'kalimera'
```

```
>>> len(a)
```

```
8
```

```
>>> a.count('a')
```

```
2
```

```
>>> a.capitalize()
```

```
'Kalimera'
```

Συνάρτηση εισόδου - *input*

Προτρέπει τον χρήστη να εισάγει δεδομένα.

Για τη σωστή εισαγωγή δεδομένων, απαιτείται η χρήση συνάρτησης η οποία θα μετατρέψει τα δεδομένα αυτά σε αριθμό για να γίνουν αριθμητικοί υπολογισμοί. (*int-float-eval*)

Συναρτήσεις int, float, eval, str

Χρήση των
συναρτήσεων
int, float και eval

σε συμβολοσειρές.

Αν μια συμβολοσειρά
αποτελείται από αριθμητική
έκφραση η **eval** επιλέγει
αυτόματα τον κατάλληλο
αριθμητικό τύπο ,ακέραιο ή
κινητής υποδιαστολής,
ανάλογα με την περίπτωση

```
>>> print(int('23'))
```

```
23
```

```
>>> print(float('23'))
```

```
23.0
```

```
>>> print(eval('23.5'))
```

```
23.5
```

```
>>> x=5
```

```
>>> print(eval('23+(2*x)'))
```

```
33
```

Παράδειγμα χρήσης float

Χρησιμοποιώντας τις συναρτήσεις **int()**, **float()**, **eval()** μπορούμε να μετατρέψουμε την έξοδο της input στον κατάλληλο τύπο.

```
print('Μετατροπή από F σε C')  
f=float(input('x=?')) #Δώστε x=4  
c=5*(f-32)/9  
print('farenait ',f,'kelsioy',c)
```

```
farenait 4.0 kelsioy -15.555555555555555555
```

Συνάρτηση εξόδου - *print*

- Χρησιμοποιείται για να εμφανίζει αριθμούς και συμβολοσειρές στην οθόνη
 - Αν n είναι ένας αριθμός, **print** (n) εμφανίζει τον αριθμό n .
 - Η συνάρτηση **print**('s'), εμφανίζει τη συμβολοσειρά 's'
 - Η συνάρτηση **print** εμφανίζει το αποτέλεσμα μιας αριθμητικής έκφρασης.
 - Η **print** μπορεί να εμφανίσει πολλές τιμές.

```
print(3.5)
```

```
print(3/6,5**2*2)
```

```
a='ha'
```

```
print('a')*3
```

run

```
3.5
```

```
0.5 50
```

```
haha
```

Προαιρετικό όρισμα: sep της Print

`print(value0, value1, ..., valueN)`

- Η συνάρτηση `print` εκτυπώνει-εμφανίζει, χωρίζοντας τις τιμές με διαστήματα.
- Μπορούμε να αλλάξουμε αυτό το διαχωρισμό χρησιμοποιώντας το όρισμα **sep**

```
>>> print('Hellow','World',sep='**')
```

HellowWorld**

```
>>> print('Hellow','World!',sep='!!!')
```

Hellow!!!World!

```
>>> print('Hellow','World!',sep='  ')
```

Hellow World!

Προαιρετικό όρισμα **end** της Print

Το τμήμα *end* μπορεί να συμπεριλάβει στα «» οτιδήποτε είναι επιθυμητό να εκτυπώνεται στο τέλος κάθε εντολής `print`, καταργώντας τη δημιουργία νέας γραμμής (για αυτήν την `print`)

```
>>> print('Hellow',end="")
```

Hellow

```
>>> print('World')
```

World

```
>>> (δημιουργία νέας γραμμής)
```

Χρήση χαρακτήρων διαφυγής

- Μια μεγάλη πρόταση μπορεί να διαιρεθεί σε δύο ή περισσότερες γραμμές.
 - Κάθε γραμμή τελειώνει με τον χαρακτήρα (\)
- Οποιοσδήποτε κώδικας περικλείεται μέσα σε παρενθέσεις μπορεί να εκτείνεται σε πολλές γραμμές.
- Για εμφάνιση αποτελεσμάτων στην ίδια γραμμή.
- \t εισάγει ένα νέο στηλοθέτη
- \n εισάγει μια νέα γραμμή

Παραδείγματα

- Στην Python, ο χαρακτήρας διαφυγής `\n` χρησιμοποιείται για να προσθέσει μια νέα γραμμή (line break) μέσα σε μια συμβολοσειρά.

Παράδειγμα χρήσης του `\n`:

```
>>> print("Hello\nWorld")
```

Hello

World

- Στην Python ο χαρακτήρας διαφυγής `\t` χρησιμοποιείται για να εισαγάγει έναν οριζόντιο tab σε μία συμβολοσειρά.

Παράδειγμα της χρήσης του `\t`:

```
>>> print('Hellow\tWorld!')
```

Hellow World!

Χρήση πολλαπλών `\t`

```
>>> print('a\tb\tc')
```

a

b

c

Τρόποι μορφοποίησης εξόδου

1. Με την χρήση Μορφοποίησης Υπολοίπου (%)

ΑΚΕΡΑΙΑ ΜΕΤΑΒΛΗΤΗ	
%d	Ακέραιος αριθμός (int)
%wd	Ακέραιος αριθμός με πλάτος πεδίου w
ΑΛΦΑΡΙΘΜΗΤΙΚΗ ΜΕΤΑΒΛΗΤΗ	
%s	Αλφαριθμητικό (str)
%c	χαρακτήρας
ΠΡΑΓΜΑΤΙΚΗ ΜΕΤΑΒΛΗΤΗ	
%f	Πραγματικός αριθμός με 6 δεκαδικά ψηφία
%.2f	Πραγματικός αριθμός με 2 δεκαδικά ψηφία
%w.zf	Πραγματικός αρ. με πλάτος w εκ των οποίων z δεκαδικά ψηφία

Παραδείγματα

```
>>> per=7
```

```
>>> art=6
```

```
>>> print('Έδωσες %d περιτούς και %d άρτιους'%(per,art))
```

```
Έδωσες 7 περιτούς και 6 άρτιους
```

```
>>> print('Τεμάχια:%d Τιμή:%d ΦΠΑ:%.2f'%(22,10,24))
```

```
Τεμάχια:22 Τιμή:10 ΦΠΑ:24.00
```

2.Στοιχίση Εξόδου σε Πεδίο

Παράδειγμα: `ljust(n)`, `rjust(n)`, και `center(n)` μπορούν να χρησιμοποιηθούν για να στοιχίσουν αριστερά, δεξιά και κέντρο.

Παράδειγμα:

```
>>> print('012345678901234567890123456789')
```

1ο πεδίο

2ο πεδίο

3ο πεδίο

```
012345678901234567890123456789
```

Σε ένα πεδίο 5 θέσεων, τύπωσε με αριστερή στοιχίση τη λέξη 'Τάξη', μετά σε ένα πεδίο 20 θέσεων τύπωσε με αριστερή στοιχίση τη λέξη Παίκτης. Στη συνέχεια σε ένα πεδίο 5 θέσεων τύπωσε με στοιχίση στο κέντρο τους χαρακτήρες GR με κανένα διαχωριστικό τα πεδία μεταξύ τους(`sep=""`).

```
>>> print('Τάξη'.ljust(5),'Παίκτης'.ljust(20),'GR'.rjust(5),sep="")
```

```
012345678901234567890123456789
```

Τάξη Παίκτης

GR

2. Συνέχεια παραδείγματος

```
>>> print('Τάξη'.ljust(5),'Παίκτης'.ljust(20),'GR'.rjust(3),sep='')
```

```
012345678901234567890123456789
```

```
Τάξη Παίκτης GR
```

Σε ένα πεδίο 5 θέσεων τυπώνουμε με στοίχισή στο κέντρο τον αριθμό 1. Στη συνέχεια σε ένα πεδίο 20 θέσεων τυπώνουμε με αριστερή στοίχιση το ονοματεπώνυμο, Γεωργίου Νίκος, και τέλος σε ένα πεδίο 5 θέσεων με στοίχιση στο κέντρο τον αριθμό 62 με κανένα διαχωριστικό ανάμεσα στα πεδία.

```
>>> print('1'.center(5),'Γεωργίου Νίκος'.ljust(20),'62'.rjust(5),sep='')
```

```
1 Γεωργίου Νίκος 62
```

Και όλα μαζί:

```
0123456789012345678901234567
```

```
Τάξη Παίκτης GR
```

```
1 Γεωργίου Νίκος 62
```

Με τον ίδιο τρόπο:

```
2 Λάμπρου Θωμάς 55
```

```
3 Δήμου Μαρία 42
```

3.Στοίχιση Εξόδου με format

Η μέθοδος format μπορεί να τοποθετήσει διαχωριστικές χιλιάδες, να στρογγυλεύει αριθμούς και να μετατρέπει αριθμούς σε ποσοστά.

```
>>> print('{0:<20}'.format('Αύριο')) <=αριστερή στοίχιση
```

```
Αύριο
```

```
>>> print('{0:^20}'.format('Αύριο'))
```

^=στοίχιση στο κέντρο

```
Αύριο
```

```
>>> print('{0:>20}'.format('Αύριο'))
```

>=δεξιά στοίχιση

```
Αύριο
```

ΙΣΟΔΥΝΑΜΗ ΣΤΟΙΧΙΣΗ

```
>>> print('ayrio'.ljust(20))
```

```
ayrio
```

```
>>> print('ayrio'.center(20))
```

```
ayrio
```

```
print('ayrio'.rjust(20))
```

```
ayrio
```


Μορφοποίηση αριθμού

Εντολή	Έξοδος	Ιδιότητα
<code>print('{0:10d}'.format(12345678))</code>	12345678	ακέραιος
<code>print('{0:10,d}'.format(12345678))</code>	12,345,678	Με διαχωριστικό χιλιάδων
<code>print('{0:10.2f}'.format(1234.5678))</code>	1234.57	2 δεκαδικά και στρογγυλοποίηση
<code>print('{0:10,.2f}'.format(1234.5678))</code>	1,234.57	Κόμμα στις χιλιάδες τελεία στις δεκάδες και στρογγυλοποίηση
<code>print('{0:10.2%}'.format(12.345678))</code>	1234.57%	Προσθέτει %, και στρογγυλοποίηση

Στοιχίση Εξόδου με format (Παράδειγμα 2^ο)

```
print('{0:>5}{1:^15}{2:<15}{3:>10}'.format('A/A', 'Όνομα', 'Επίθετο', 'Ημερομηνία  
Γέννησης'))
```

A/A Όνομα Επίθετο Ημερομηνία Γέννησης

```
print('{0:>5}{1:^15}{2:<15}{3:>10}'.format('1', 'Νικόλαος', 'Γεωργίου', '12/03/2004'))
```

1 Νικόλαος Γεωργίου 12/03/2004

>>>

```
print('{0:>5}{1:^15}{2:<15}{3:>10}'.format('2', 'Μαρία', 'Δουγαλή', '22/09/2005'))
```

2 Μαρία Δουγαλή 22/09/2005

Μέσα στο άγγιστρο ο 1^{ος} αριθμός προσδιορίζει τον αριθμό του ορίσματος της format: 0 για το 1^ο, 1 για το 2^ο, 2 για το 3^ο κ.ο.κ.

Για απλές μορφοποιήσεις με τη format και το ισοδύναμό τους με άλλο, str='ο' τρόπο (w, το μήκος του πεδίου, num=αριθμός)

```
print('{0:<w}'.format(num))
```

```
print(str(num).ljust(w))
```

```
print('{0:<w}'.format(num))
```

```
print(str(num).center(w))
```

```
print('{0:>w}'.format(num))
```

```
print(str(num).rjust(w))
```

Συγκεντρωτικά παραδείγματα

```
print('{:25}'.format('abrakatabra'))
```

```
print('{:25s}'.format('abrakatabra'))#ίδιο αποτέλεσμα
```

```
print('abrakatabra'.ljust(25)) # “
```

```
print('0|234567890|234567890|23456')
```

```
print('{:>25}'.format('abrakatabra'))
```

```
print('abrakatabra'.rjust(25))
```

```
print('{:^55}'.format('abrakatabra'))
```

```
print('{:<25}'.format('abrakatabra'))
```

```
metabliti=2.123456789
```

```
print('{0:<7}{1:^20}{2:>9}{3:9,.4f}'.format('abra','kat','abra',metabliti))
```

```
abra      kat      abra 2.1235
```

```
print('0|234567890|234567890|234567890|23456789')
```

```
print('{0:^9}{1:<20}{2:>3}'.format('Rank','Player','HR'))
```

```
print('Rank'.ljust(10),'Player'.center(20),'HR'.ljust(7),sep='*')
```

```
a=10
```

```
b=11
```

```
print('Ο αριθμός {0:.2f} είναι μικρότερος από τον αριθμό {1:d}'.format(a,b))
```

```
print('{0:.2f}'.format(1.123))
```

Χωρίζοντας το Ονοματεπώνυμο!!!

Σπάμε το ονοματεπώνυμο σε 2 τμήματα, Το Όνομα και το Επώνυμο

```
fullname=input('Εισήγαγε πρώτα το όνομα, μετά το  
κενό, και τέλος επώνυμο : ').
```

Αν το όνομα καταλαμβάνει n θέσεις, το κενό
βρίσκεται στη θέση $n+1$.

```
n=fullname.find(' ') #Το κενό σημαίνει ότι θα αρχίσει  
το Επίθετο
```

```
print("Όνομα: ',fullname[:n])
```

```
print('Επίθετο: ',fullname[n+1])
```

Εισήγαγε το Ονοματεπώνυμο: Ανδρέας Νικολάου

Όνομα: Ανδρέας

Επίθετο: Νικολάου