



# Συναρτήσεις

# Συναρτήσεις

- ▶ Η δημιουργία μιας συνάρτησης γίνεται με `def`. Επιπλέον, της ανατίθεται και ένα όνομα.
- ▶ Το `return` επιστρέφει μια τιμή στον καλούντα.
- ▶ Τα ορίσματα περνούν με ανάθεση τιμής.
- ▶ Οι τύποι των ορισμάτων και των επιστρεφόμενων τιμών δεν δηλώνονται.

```
def <name>(arg1, arg2, ..., argN):  
    <statements>  
    return <value>
```

```
def times(x, y):  
    return x*y
```

# Πέρασμα ορισμάτων σε συναρτήσεις

- ▶ Τα ορίσματα περνούν με ανάθεση.
- ▶ Τα ορίσματα ανατίθενται σε τοπικά ονόματα.
- ▶ Η ανάθεση στα ορίσματα δεν επηρεάζει τον καλούντα.
- ▶ Αλλάζοντας ένα μεταβλητό όνομα μπορεί να επηρεάσει τον καλούντα.

```
def changer (x, y):  
    x = 2                # changes local value of x only  
    y[0] = 'hi'         # changes shared object
```

# Προεπιλεγμένες τιμές ορισμάτων

- ▶ Μπορούν να οριστούν προεπιλεγμένες τιμές για ορίσματα που πρέπει να περαστούν σε μια κλήση συνάρτησης.

```
def func(a, b, c=10, d=100):  
    print (a, b, c, d)
```

```
>>> func(1,2)  
1 2 10 100
```

```
>>> func(1,2,3,4)  
1,2,3,4
```



# Πέρασμα παραμέτρων με βάση το όνομα των τυπικών παραμέτρων

```
def func(a, b, c, d):  
    print (a, b, c, d)  
  
>>> func(b=2, a=1, d=100, c=10)  
  
1 2 10 100
```

# Συλλογή Θεσιακών παραμέτρων

```
>>> def provisions(necessary, useful, *helpful) #Προσοχή στο σύμβολο '*'
    print('Απαραίτητο είναι το: ', necessary)
    print('Χρειάζεται και το: ', useful)
    print('Καλό είναι να έχω και: ', helpful)
```

```
>>> provisions('Παλτό', 'Κασκόλ', 'Γάντια', 'Κιάλια', 'Καπέλο')
```

Απαραίτητο είναι το: Παλτό

Χρειάζεται και το: Κασκόλ

Καλό είναι να έχω και: ('Γάντια', 'Κιάλια', 'Καπέλο')

- Γίνεται ομαδοποίηση των θεσιακών παραμέτρων σε μια πλειάδα τιμών.

# Συλλογή ονοματισμένων παραμέτρων

- ▶ Μπορούν να περαστούν ονοματισμένες παράμετροι με τη μορφή λεξικού με χρήση του συμβόλου '\*\*'.

```
>>> def countries_letters(**kwargs):
```

```
    print('Παράμετροι', kwargs)
```

```
>>> countries_letters(gr='Ελλάδα', en='Μεγάλη Βρετανία', fr='Γαλλία')
```

```
Παράμετροι {'fr': 'Γαλλία', 'en': 'Μεγάλη Βρετανία', 'gr': 'Ελλάδα'}
```



# Τεχνικές περάσματος παραμέτρων

- ▶ Υπάρχουν δύο τεχνικές περάσματος παραμέτρων σε συναρτήσεις:
  - ▶ Πέρασμα με τιμή (call by value)
  - ▶ Πέρασμα με αναφορά (call by reference)
- ▶ Όταν ο τύπος της παραμέτρου είναι αμετάβλητος τότε το πέρασμα γίνεται με τιμή.
- ▶ Όταν ο τύπος της παραμέτρου είναι μεταβλητός τότε το πέρασμα γίνεται με αναφορά.
- ▶ Κατά το πέρασμα με τιμή οι αλλαγές στην τιμή μιας τυπικής παραμέτρου δεν επηρεάζουν την πραγματική παράμετρο στην καλούσα συνάρτηση.
- ▶ Κατά το πέρασμα με αναφορά οι αλλαγές στην τιμή μιας τυπικής παραμέτρου επηρεάζουν την πραγματική παράμετρο στην καλούσα συνάρτηση.



# Παράδειγμα περάσματος με αναφορά

```
>>> langs_list=['C', 'perl', 'Java', 'php']
```

```
>>> def change3rd(x):
```

```
    x[2]="Python"
```

```
>>> change3rd(langs_list)
```

```
>>> langs_list
```

```
['C', 'perl', 'Python', 'php']
```

# Μερικές σημαντικές λεπτομέρειες

- ▶ Όλες οι συναρτήσεις έχουν επιστρεφόμενη τιμή, ακόμη κι όταν δεν υπάρχει γραμμή return εντός του κώδικα.
- ▶ Οι συναρτήσεις χωρίς γραμμή return επιστρέφουν την ειδική τιμή None.
- ▶ Στην Python δεν μπορούν δύο συναρτήσεις να έχουν το ίδιο όνομα, ακόμη κι αν έχουν διαφορετικά ορίσματα.
- ▶ Οι συναρτήσεις μπορούν να χρησιμοποιηθούν όπως κάθε άλλος τύπος δεδομένων. Έτσι μπορούν να είναι:
  - ▶ Ορίσματα σε συναρτήσεις
  - ▶ Επιστρεφόμενες τιμές συναρτήσεων
  - ▶ Να ανατεθούν σε μεταβλητές
  - ▶ Μέρη από πλειάδες, λίστες, κλπ.

# Επιστροφή πολλών τιμών

- ▶ Σύνταξη: `return <value1>, <value2>, <value3>, ...`
- ▶ Οι επιστρεφόμενες τιμές μπορούν να ανατεθούν σε αντίστοιχες μεταβλητές ή σε πλειάδα.

```
>>> def operations(a,b):  
    add_result=a+b  
    sub_result=a-b  
    return add_result, sub_result
```

```
>>> add, sub=operations(45,5)
```

```
>>> add
```

```
50
```

```
>>> sub
```

```
40
```

```
>>> operations(45,5)
```

```
(50,40)
```



# Εμβέλεια μεταβλητών



- ▶ Τοπική μεταβλητή: είναι προσπελάσιμη μόνο από το εσωτερικό μιας συνάρτησης.
  - ▶ Μια τοπική μεταβλητή έχει διάρκεια ζωής ίση με τη διάρκεια ζωής της συνάρτησης.
  - ▶ Όταν ολοκληρωθεί η συνάρτηση οι τοπικές της μεταβλητές καταστρέφονται και οι τιμές τους απορρίπτονται.
- ▶ Καθολική μεταβλητή (global variable): ορίζεται εντός ενός δομικού στοιχείου, (module), αλλά εκτός οποιασδήποτε συνάρτησης.
- ▶ Αν μέσα σε μια συνάρτηση χρησιμοποιείται μια τοπική μεταβλητή που έχει το ίδιο όνομα με μια καθολική μεταβλητή, τότε οι αλλαγές των τιμών σε αυτό το όνομα γίνονται στην τοπική μεταβλητή.
- ▶ Για να αναφερθούμε μέσα από μια συνάρτηση σε μια καθολική μεταβλητή πρέπει πριν το όνομα της μεταβλητής να γράψουμε τη δήλωση global.



# Η δήλωση global

```
>>> global_var=5
>>> def manip_global():
    global global_var
    global_var+=1
>>> manip_global()
>>> global_var
```

6