In [1]:
```
students = [
    'student 1',
    'student 2',
]

import random
def pick():
    return random.choice(students)
```

In [2]:
```
pick()
```

Out[2]: 'Αθανασία'

Functions

In [3]:
```
def f(x):
    return x+1
```

In [7]:
```
pick()
```

Out[7]: 'Αιμίλιος'

In [5]:
```
f(5)
```

Out[5]: 6

In [6]:
```
def f2():
    return 42
```

In [8]:
```
f2()
```

Out[8]: 42

In [9]:
```
def f3(x,y):
    return x+y
```

In [12]:
```
pick()
```

Out[12]: 'Ελσα'

In [ ]:
```

```

In [11]:
```
f3(2,3)
```

Out[11]: 5

In [13]:
```
f3(2,f2())
```

Out[13]: 44

In [14]:
```
def f4(a,b,c,d,e,f,g,h):
    return a-b
```

In [15]:
```
pick()
```

Out[15]: 'Κλάρα'

In [17]:
```
f4(1,2,3,4,5,7,8,9)
```

Out[17]: -1

In [18]:
```
def f5(a,b):
    c=a+b
```

In [21]:
```
pick()
```

Out[21]: 'Αθανασία'

In [20]:
```
f5(4,5)
```

In [22]:
```
a=f5(4,5)
print (a)
```

```
None
```

In [25]:
```
a=3
def f():
    a=4
f()
print (a)
```

```
3
```

In [28]:
```
a = [1,2,3]
def f():
    a.append(4)
f()
print (a)
```

```
[1, 2, 3, 4]
```

In [29]:
```
b = [1,2,3]
b += [4]
print (b)
```

```
[1, 2, 3, 4]
```

In [30]:
```
b
```

Out[30]: [1, 2, 3, 4]

In [31]:
```
b.append(6)
b
```

Out[31]: [1, 2, 3, 4, 6]

In [33]:
```
a = [1,2,3]
def f():
    #print (a)
    a = a + [4]
f()
```

```
---------------------------------------------------------------------------
UnboundLocalError                         Traceback (most recent call last)
<ipython-input-33-fa1626706d0c> in <module>()
      3     #print (a)
      4     a = a + [4]
----> 5 f()

<ipython-input-33-fa1626706d0c> in f()
      2 def f():
      3     #print (a)
----> 4     a = a + [4]
      5 f()

UnboundLocalError: local variable 'a' referenced before assignment
```

In [38]:
```
a=4
def f():
    print (a)
f()
```

```
4
```

In [39]:
```
a=4
def f():
    a=3
    print (a)
f()
print (a)
```

```
3
4
```

BAD PRACTICE

1. Μην κάνετε access μεταβλητές που είναι ορισμένες έξω από τη συνάρτηση
2. Μην χρησιμοποιείτε τα ίδια ονόματα μεταβλητών μέσα σε μία συνάρτηση με έξω

In [40]:
```
a ='q/w,kdfhal,sdjkfhalsufghalkudfghalkduhfgaldkfghlkjfghlkxdjfghlsdkjfghlskjfgh'

def length_str(x):
    return len(x)

length_str(a)
```

Out[40]: 76

In [42]:
```
a ='q/w,kdfhal,sdjkfhalsufghalkudfghalkduhfgaldkfghlkjfghlkxdjfghlsdkjfghlskjfgh'

def length_str():
    return len(a)

length_str()
```

Out[42]: 76

In [43]:
```
a ='q/w,kdfhal,sdjkfhalsufghalkudfghalkduhfgaldkfghlkjfghlkxdjfghlsdkjfghlskjfgh'

def length_str():
    a = a + '123123123'
    return len(a)

length_str()
```

```
---------------------------------------------------------------------------
UnboundLocalError                         Traceback (most recent call last)
<ipython-input-43-6e6cb4b259f1> in <module>()
      5     return len(a)
      6
----> 7 length_str()

<ipython-input-43-6e6cb4b259f1> in length_str()
      2
      3 def length_str():
----> 4     a = a + '123123123'
      5     return len(a)
      6

UnboundLocalError: local variable 'a' referenced before assignment
```

In [45]:
```
a ='q/w,kdfhal,sdjkfhalsufghalkudfghalkduhfgaldkfghlkjfghlkxdjfghlsdkjfghlskjfgh'

def length_str(b):
    a = b + '123123123'
    return len(a)

length_str('12121')
```

Out[45]:
```
14
```

In [46]:
```
def f7():
    return 1,2
```

In [47]:
```
a,b = f7()
```

In [48]:
```
print (a)
```

```
1
```

In [49]:
```
print (b)
```

```
2
```

In [50]:
```
def f(x):
    return x.upper(), x.lower()

f('Mitsos')
```

Out[50]:
```
('MITSOS', 'mitsos')
```

In [51]:
```
def f(x):
    return x.upper(), x.lower()

a,b = f('Mitsos')
```

In [52]:
```
print (a)
```

```
MITSOS
```

In [53]:
```
print (b)
```

```
mitsos
```

tuples

In [54]:
```
a = [1,2,3,4]
```

In [55]:
```
b = (1,2,3,4)
```

In [56]:
```
print (a)
```

```
[1, 2, 3, 4]
```

In [57]:
```
print (b)
```

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

In [58]:
```
a += [5]
```

In [59]:
```
print (a)
```

```
[1, 2, 3, 4, 5]
```

In [61]:
```
b += (5)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-61-8a50bc65f783> in <module>()
----> 1 b += (5)

TypeError: can only concatenate tuple (not "int") to tuple
```

In [62]:
```
def f8():
    return 1,2
```

In [63]:
```
a=f8()
print (a)
```

```
(1, 2)
```

In [64]:
```
a[0]=4
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-64-998c6162715b> in <module>()
----> 1 a[0]=4

TypeError: 'tuple' object does not support item assignment
```

In [65]:
```
a = [1,2,3,4]
```

In [67]:
```
a[0]=8
```

In [68]:
```
print (a)
```

```
[8, 2, 3, 4]
```

In [69]:
```
b = (1,2,3,4)
print (b[0])
```

```
1
```

In [71]:
```
b[0] = 8
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-71-564545b08076> in <module>()
----> 1 b[0] = 8

TypeError: 'tuple' object does not support item assignment
```

In [72]:
```
a = [1,2,3,4]
tuple(a)
```

Out[72]:
```
(1, 2, 3, 4)
```

In [73]:
```
a = (1,2,3,4)
list(a)
```

Out[73]:
```
[1, 2, 3, 4]
```

In [74]:
```
f8()
```

Out[74]:
```
(1, 2)
```

In [75]:
```
list(f8())
```

Out[75]:
```
[1, 2]
```

In [102]:
```
def f():
    def g(x):
        return x+3
    return g
```

In [101]:
```
def g(x):
    return x+3
g(1)
```

Out[101]:
```
4
```

In [95]:
```
def f():
    g = [1,2,4]
    return g
```

In [90]:
```
f()
```

Out[90]:
```
<function __main__.f.<locals>.g(x)>
```

In [ ]:
```
f()(4)
```

In [93]:
```
def f():
    return 42
```

In [94]:
```
type(f)
```

Out[94]:
```
function
```

In [104]:
```
def f(x):
    return x+3
```

In [105]:
```
g=f
```

In [107]:
```
g(5)
```

Out[107]:
```
8
```

In [119]:
```
def f(x,y):
    def g(z):
        return x+y+z
    return g
```

In [ ]:
```

```

In [ ]:
```
f(1,2)
#Returns:
#def g(z):
#    return 1+2+z
```

In [121]:
```
k=f(1,2)
print (k)
```

```
<function f.<locals>.g at 0x10c1e9f28>
```

In [117]:
```
k(4)
```

Out[117]:
```
7
```

In [118]:
```
f(1,2)(4)
```

Out[118]:
```
7
```

In [122]:
```
def f(x,y):
    def g(z):
        return x+y+z
    return g
```

In [123]:
```
a = [f(1,2), f(3,4), f(4,8)]
```

In [130]:
```
a[1](4)
```

Out[130]:
```
11
```

In [131]:
```
def g(z):
    return 3+4+z
g(4)
```

Out[131]:
```
11
```

In [133]:
```
def f(a, *b):
    print(a)
    print(b)

f(1)
```

```
1
()
```

In [141]:
```
f(1,2)
```

```
1
(2,)
```

In [142]:
```
f(1,2,3,4,5,6,7,8,)
```

```
1
(2, 3, 4, 5, 6, 7, 8)
```

In [ ]:
```

```

In [138]:
```
a = [1]
a
```

Out[138]:
```
[1]
```

In [139]:
```
a = (1,)
```

In [140]:
```
a
```

Out[140]:
```
(1,)
```

In [144]:
```
def f(a,b,c,d):
    return a+b+c+d
```

In [145]:
```
f(4,5,3,2)
```

Out[145]:
```
14
```

In [146]:
```
a = [4,5,3,2]
```

In [147]:
```
f(*a)
```

Out[147]:
```
14
```

In [148]:
```
a = (1,2,3,4)
f(*a)
```

Out[148]:
```
10
```

In [149]:
```
a = (1,2,3,4,5)
f(*a)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-149-2d89997f47c4> in <module>()
      1 a = (1,2,3,4,5)
----> 2 f(*a)

TypeError: f() takes 4 positional arguments but 5 were given
```

In [156]:
```
def f(a,b,c,d,*e):
    # Calculate sum of e
    s = 0
    i = 0
    print (e)
    while i<len(e):
        s += e[0]
        i +=1
    print (a+b+c+d+s)
```

In [154]:
```
f(1,2,3,4,5,6)
```

```
20
```

In [157]:
```
a = [1,2,3,4,5,6]

f(4,5,*a)

f(4,5,1,2,3,4,5,6)
```

```
(3, 4, 5, 6)
24
```

In [ ]:
```
def f(a,b,c,d,*e):
    # Calculate sum of e
    pr
    print (a+b+c+d+e)
```

In [162]:
```
(1,2) + (1,2) + ('zfsdfsdf',)
```

Out[162]:
```
(1, 2, 1, 2, 'zfsdfsdf')
```

In [165]:
```
def f():
    return 42
def g(x):
    return x+4
def h(x,y):
    return x+y
def i(x):
    return x.upper(), x.lower()
def j():
    c = 42
```

In [166]:
```
def f(a,b=4):
    return a+b
```

In [168]:
```
f(3)
```

Out[168]:
```
7
```

In [171]:
```
f(5,8)
```

Out[171]:
```
13
```

for

In [172]:
```
for x in [2,3,5]:
    print (x)
```

```
2
3
5
```

In [173]:
```
for x in (2,3,5):
    print (x)
```

```
2
3
5
```

In [174]:
```
for x in 'mitsos':
    print (x)
```

```
m
i
t
s
o
s
```

In [175]:
```
a=[3,4,5,6]
```

In [176]:
```
sum(a)
```

Out[176]: 18

In [177]:
```
min(a)
```

Out[177]: 3

In [178]:
```
max(a)
```

Out[178]: 6

In [179]:
```
len(a)
```

Out[179]: 4

In [180]:
```
a.count(4)
```

Out[180]: 1

In [183]:
```
a.index(5)
```

Out[183]: 2

In [185]:
```
a=[3,4,5,6]
a.count(10)
```

Out[185]: 0

In [187]:
```
a.index(10)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-187-be2bb036fbfd> in <module>()
----> 1 a.index(10)

ValueError: 10 is not in list
```

In [188]:
```
a=[4,5,6,4]
a.index(4)
```

Out[188]: 0

In [ ]:
```

```

In [182]:
```
sum(a)/len(a)
```

Out[182]: 4.5

List comprehensions

In [192]:
```
a=[4,5,6,4]
i = 0
apotelesma = []
for x in a:
    #i += 1
    if x==4:
        apotelesma += [i]
    i += 1
print (apotelesma)
```

```
[0, 3]
```

In [195]:
```
a=[4,5,6,4]
# [8, 10, 12, 8]
```

In [197]:
```
for x in a:
    print (x*2)
```

```
8
10
12
8
```

In [198]:
```
b = []
for x in a:
    b.append(x*2)
print (b)
```

```
[8, 10, 12, 8]
```

In [207]:
```
%%timeit
b = []
i = 0
while i<len(a):
    b.append(2*a[i])
    i += 1
#print (b)
```

```
787 ns ± 9.54 ns per loop (mean ± std. dev. of 7 runs, 1000000 loops each)
```

In [208]:
```
%%timeit
b = []
for x in a:
    b.append(x*2)
#print (b)
```

```
367 ns ± 3.94 ns per loop (mean ± std. dev. of 7 runs, 1000000 loops each)
```

In [209]:
```
%%timeit
[x*2 for x in a]
```

```
308 ns ± 5.14 ns per loop (mean ± std. dev. of 7 runs, 1000000 loops each)
```

In [205]:
```
[x/2 for x in a]
```

Out[205]:
```
[2.0, 2.5, 3.0, 2.0]
```

In [210]:
```
a=[4,5,6,4]

# DIplasia mono gia ta artia
```

In [224]:
```
%%timeit
i=0
apotelesma = []
while i<len(a):
    if a[i] % 2 == 0:
        apotelesma.append(a[i]*2)
    i+=1
apotelesma
```

```
919 ns ± 7.68 ns per loop (mean ± std. dev. of 7 runs, 1000000 loops each)
```

In [225]:
```
%%timeit
apotelesma = []
for x in a:
    if x%2 == 0:
        apotelesma.append(2*x)
apotelesma
```

```
454 ns ± 15.6 ns per loop (mean ± std. dev. of 7 runs, 1000000 loops each)
```

In [226]:
```
%%timeit
[x*2 for x in a if x%2==0]
```

```
426 ns ± 5.78 ns per loop (mean ± std. dev. of 7 runs, 1000000 loops each)
```

In [221]:
```
b=[x*2 for x in a if not x%2]
```

In [222]:
```
[0] + [x*2 for x in a if not x%2]
```

Out[222]:
```
[0, 8, 12, 8]
```

In [223]:
```
for y in [x*2 for x in a if not x%2]:
    print (y*3)
```

```
24
36
24
```

In [228]:
```
a = [4,5,6]
b = [10,20]

[x*y for x in a for y in b]
```

Out[228]:
```
[40, 80, 50, 100, 60, 120]
```

In [230]:
```
apotelesma = []
for x in a:
    for y in b:
        apotelesma.append(x*y)
apotelesma
```

Out[230]:
```
[40, 80, 50, 100, 60, 120]
```

In [231]:
```
apotelesma = []
i=0
while i<len(a):
    j=0
    while j<len(b):
        apotelesma.append(a[i]*b[j])
        j += 1
    i += 1
apotelesma
```

Out[231]: [40, 80, 50, 100, 60, 120]

Kolpakia

1. Μέτρημα

In [233]:
```
a = [3,4,5,4,3,2,3]
```

In [234]:
```
counter = 0
for x in a:
    if x%2==0:
        counter += 1
counter
```

Out[234]: 3

In [236]:
```
sum([1 for x in a if x%2==0])
```

Out[236]: 3

In [238]:
```
[x*2 for x in a if x%2==0]
```

Out[238]: [8, 8, 4]

In [240]:
```
[x for x in a if x%2==0]
```

Out[240]: [4, 4, 2]

In [242]:
```
['nitsa' for x in a if x%2==0]
```

Out[242]: ['nitsa', 'nitsa', 'nitsa']

In [244]:
```
sum([1 for x in a if x%2==0])
```

Out[244]: 3

In [246]:
```
len(['nitsa' for x in a if x%2==0])
```

Out[246]: 3

In [250]:
```
[x%2==0 for x in a]
```

Out[250]: [False, True, False, True, False, True, False]

In [251]:
```
sum([x%2==0 for x in a])
```

Out[251]: 3

In [252]:
```
103 % 2 == 0
```

Out[252]:
```
False
```

In [ ]:
```
103 % 2
```

In [253]:
```
pick()
```

Out[253]:
```
'Άρτεμις'
```

In [254]:
```
islands = [
    "Astypalaia",
    "Kalymnos",
    "Karpathos",
    "Kasos",
    "Kastellorizo",
    "Kos",
    "Leros",
    "Nisyros",
    "Patmos",
    "Rhodos",
    "Symi",
    "Tilos",]
```

In [255]:
```
students
```

Out[255]:
```
['Κλάρα',
 'Νίκος',
 'Άρτεμις',
 'Ιπποκράτης',
 'Πολίνα',
 'Δανάη',
 'Αθανασία',
 'Αιμίλιος',
 'Έλσα',
 'Γιάννης']
```

In [257]:
```
[x[0] in ['A', 'I', 'E', 'O']   for x in students]
```

Out[257]:
```
[False, False, False, True, False, False, True, True, False, False]
```

In [260]:
```
sum([x[0] in ['A', 'I', 'E', 'O', 'E', 'A']   for x in students])
```

Out[260]:
```
5
```

In [267]:
```
[x for x in students if x[0] in ['A', 'I', 'E', 'O', 'E', 'A']]
```

Out[267]:
```
['Άρτεμις', 'Ιπποκράτης', 'Αθανασία', 'Αιμίλιος', 'Έλσα']
```

In [268]:
```
[x.upper() for x in students if x[0] in ['A', 'I', 'E', 'O', 'E', 'A']]
```

Out[268]:
```
['ΑΡΤΕΜΙΣ', 'ΙΠΠΟΚΡΑΤΗΣ', 'ΑΘΑΝΑΣΙΑ', 'ΑΙΜΙΛΙΟΣ', 'ΕΛΣΑ']
```

In [265]:
```
[print(x) for x in students if x[0] in ['A', 'I', 'E', 'O', 'E', 'A']]
```

```
Άρτεμις
Ιπποκράτης
Αθανασία
Αιμίλιος
Έλσα
```

Out[265]:
```
[None, None, None, None, None]
```

In [266]:
```
a = print('Mitsos')
print (a)
```

```
Mitsos
None
```

In [270]:
```
islands
```

Out[270]:
```
['Astypalaia',
 'Kalymnos',
 'Karpathos',
 'Kasos',
 'Kastellorizo',
 'Kos',
 'Leros',
 'Nisyros',
 'Patmos',
 'Rhodos',
 'Symi',
 'Tilos']
```

In [277]:
```
[ [island.count(fonien) for fonien in ['a', 'o', 'i', 'e', 'y']] for island in is
```

Out[277]:
```
[[3, 0, 1, 0, 1],
 [1, 1, 0, 0, 1],
 [2, 1, 0, 0, 0],
 [1, 1, 0, 0, 0],
 [1, 2, 1, 1, 0],
 [0, 1, 0, 0, 0],
 [0, 1, 0, 1, 0],
 [0, 1, 1, 0, 1],
 [1, 1, 0, 0, 0],
 [0, 2, 0, 0, 0],
 [0, 0, 1, 0, 1],
 [0, 1, 1, 0, 0]]
```

In [280]:
```
[ sum([island.count(fonien) for fonien in ['a', 'o', 'i', 'e', 'y']])==1 for isla
```

Out[280]:
```
[False,
 False,
 False,
 False,
 False,
 True,
 False,
 False,
 False,
 False,
 False,
 False]
```

In [282]:
```
[island for island in islands if sum([island.count(fonien) for fonien in ['a', 'o
```

Out[282]:
```
['Kos']
```

In [284]:
```
[island for island in islands if len(island)==5 ]
```

Out[284]:
```
['Kasos', 'Leros', 'Tilos']
```

Έστω 2 strings. Ελέγξτε αν όλα τα γράμματα του πρώτου υπάρχουν στο δεύτερο.

In [296]:
```
a = 'mpanana'
b = 'mpania'

#c = [x in b for x in a]
#c.count(True) == len(c)
#not False in c
#sum(c) == len(c)
all([x in b for x in a])
```

Out[296]: True

In [312]:
```
a = 'mpanana'
b = 'pania'

#c = [x in b for x in a]
#c.count(True) == len(c)
#not False in c
#sum(c) == len(c)
all([x in b for x in a])
```

Out[312]: False

In [ ]:
```

```

In [298]:
```
sum([True, False, True])
```

Out[298]: 2

In [299]:
```
min([True, False, True])
```

Out[299]: False

In [300]:
```
max([True, False, True])
```

Out[300]: True

In [301]:
```
all([True, False, True])
```

Out[301]: False

In [302]:
```
any([True, False, True])
```

Out[302]: True

In [303]:
```
any([False, False, False])
```

Out[303]: False

In [304]:
```
any([False, False, True])
```

Out[304]: True

In [305]:
```
all([True, True, True])
```

Out[305]: True

In [306]:
```
all([True, True, False])
```

Out[306]: False

In [307]:
```
all([0, 1, 1])
```

Out[307]: False

In [309]:
```
all(['mitsos', 'kostas', 'nitsa'])
```

Out[309]: True

In [310]:
```
any([])
```

Out[310]: False

In [311]:
```
all([])
```

Out[311]: True

Paragontiko

In [313]:
```
x=6
```

In [314]:
```
list(range(10,15))
```

Out[314]: [10, 11, 12, 13, 14]

In [316]:
```
list(range(10,15,2))
```

Out[316]: [10, 12, 14]

In [318]:
```
list(range(10))
```

Out[318]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

In [319]:
```
list(range(10,1,-1))
```

Out[319]: [10, 9, 8, 7, 6, 5, 4, 3, 2]

In [323]:
```
apotelesma = 1
c = 1
while c <= x:
    apotelesma = apotelesma * c
    c += 1
apotelesma
```

Out[323]: 720

In [325]:
```
apotelesma = 1
for a in range(1, x+1):
    apotelesma = apotelesma * a
apotelesma
```

Out[325]: 720

In [326]:
```
apotelesma = 1
for a in range(1, x+1):
    apotelesma *= a
apotelesma
```

Out[326]: 720

In [331]:
```
def fact(N):
    print (N)
    if N == 1:
        return 1
    return N * fact(N-1)
```

In [332]:
```
fact(6)
```

```
6
5
4
3
2
1
```

Out[332]: 720

MIA SYNARTHSH MPOREI NA KALESEI TON EAUTO THS!!!!

In [330]:
```
6 * 5 * 4 * 3 * 2 * 1
```

Out[330]: 720

In [322]:
```
1*2*3*4*5*6
```

Out[322]: 720