

Μαθαίνω PYTHON & Tkinter

The image displays a collage of Python code and Tkinter GUI examples. At the top, a code editor shows a script that reads numbers from a file and calculates their sum. Below this, a Tkinter window is shown with a blue background and yellow squares, with the text 'Πάτησε δεξί κλικ' (Click right) at the bottom. To the right, a blue canvas contains several red lines radiating from a single point. At the bottom, another code editor shows the Tkinter code for creating a window with a canvas and text.

```
python2.py - C:\Users\KOSTAS\Desktop\python2.py (3.7.0)
File Edit Format Run Options Window Help
sum=0
saveFile=open('G:\numbers.txt','r')
for i in range(1,51) :
    kl=saveFile.readline()
    xl=int(kl)
    sum=sum+xl
saveFile.close()
print("Το άθροισμα των ψηφίων είναι: ", sum)
```

python.py - C:\Users\KOSTAS\Desktop\python.py (3.7.0)
File Edit Format Run Options Window Help
from tkinter import *
window =Tk()
window.title("ΚΕΙΜΕΝΟ")
canvas=Canvas(width=600, height=400, bg="blue")
canvas.pack()
canvas.create_text(100,100, text="hello world" , fill="yellow" ,font="arial 25")
window.mainloop()

Μαθαίνω Python & Tkinter

ISBN: 978-618-00-1733-5

Παπαστεργίου Κωνσταντίνος

2019

Περιεχόμενα

ΚΕΦ 1. ΕΙΣΑΓΩΓΗ	3
ΚΕΦ 2. ΕΝΤΟΛΕΣ ΕΙΣΟΔΟΥ – ΕΞΟΔΟΥ	5
2.1 ΕΝΤΟΛΗ print	5
2.2 ΕΝΤΟΛΗ input	9
2.2.1 Η εντολή input().....	11
ΚΕΦ 3. ΕΝΤΟΛΕΣ ΕΠΙΛΟΓΗΣ	12
3.1.2. ΕΝΤΟΛΗ if.....	12
3.1.2. ΕΝΤΟΛΗ if elif	12
3.1.3 ΕΝΤΟΛΗ if else.....	13
3.2. Οι τελεστές and και or	14
ΚΕΦ 4. ΕΝΤΟΛΕΣ ΔΟΜΗΣ ΕΠΑΝΑΛΗΨΗΣ	16
4.1. ΕΝΤΟΛΗ while	16
4.2. ΕΝΤΟΛΗ for	19
4.2.1. ΤΕΛΕΣΤΕΣ div, mod.....	21
ΚΕΦ. 5. ΤΥΧΑΙΟΙ ΑΡΙΘΜΟΙ	24
ΚΕΦ 6. ΛΙΣΤΕΣ.....	28
ΚΕΦ 7. ΣΥΝΑΡΤΗΣΕΙΣ	33
ΚΕΦ. 8. ΚΛΑΣΕΙΣ ΚΑΙ ΑΝΤΙΚΕΙΜΕΝΑ	35
ΚΕΦ. 9. ΔΙΑΧΕΙΡΙΣΗ ΑΡΧΕΙΩΝ ΜΕ ΤΗΝ PYTHON.....	39
ΚΕΦ. 10. ΓΡΑΦΙΚΑ ΜΕ ΤΗΝ ΧΡΗΣΗ ΤΟΥ Tkinter	46
ΚΕΦ. 11. TKINTER CANVAS	58
ΚΕΦ 12. Ο χρόνος στην Python.....	81
ΚΕΦ 13. ΕΠΙΛΟΓΟΣ	84
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	85

ΚΕΦ 1. ΕΙΣΑΓΩΓΗ

Όταν το 1990 ο Ολλανδός Γκίντο Βαν Ρόσουμ δημιούργησε τη γλώσσα προγραμματισμού Python, σίγουρα δεν θα περίμενε τη τεράστια αποδοχή που θα είχε στην κοινότητα των προγραμματιστών.

Η Python είναι μια γλώσσα εύκολη στη χρήση που επιτρέπει στους προγραμματιστές να υλοποιούν κώδικα πιο απλά από άλλες παραδοσιακές γλώσσες. Η ευκολία εκμάθησης της Python την καθιστά την πλέον δημοφιλή γλώσσα στον χώρο της εκπαίδευσης. Έτσι διδάσκεται στα σχολεία, γιατί είναι προσιτή κι εύκολα κατανοητή στους μαθητές.

Στο βιβλίο αυτό θα γνωρίσουμε τη γλώσσα μέσα από απλά παραδείγματα. Θα δούμε παραδείγματα με γραφικά και με επεξεργασία αρχείων, ώστε να πάρουμε μια ιδέα από τις τεράστιες δυνατότητές της.

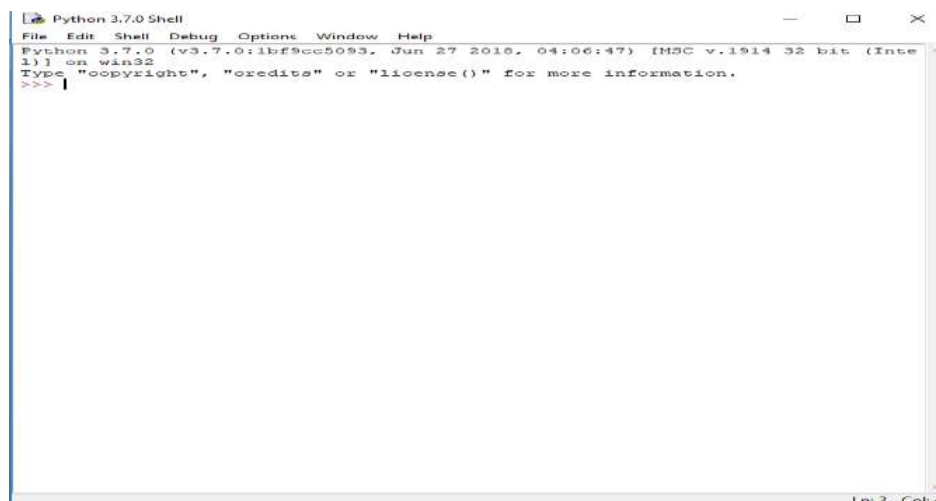
Η Python λόγω της μεγάλης δημοτικότητας που έχει αλλά και λόγω των υψηλών της δυνατοτήτων, αναμένεται να κυριαρχήσει τα επόμενα χρόνια στον χώρο του προγραμματισμού.

ΤΙ ΘΑ ΧΡΕΙΑΣΤΟΥΜΕ

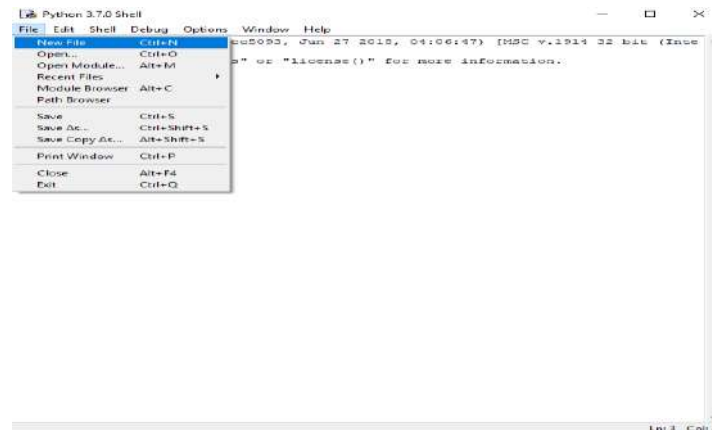
- Κατεβάζουμε την τελευταία έκδοση της Python: <https://www.python.org/downloads/>

Αφού κατεβάσουμε το αρχείο, κάνουμε εγκατάσταση ακολουθώντας τις απλές οδηγίες της εγκατάστασης.

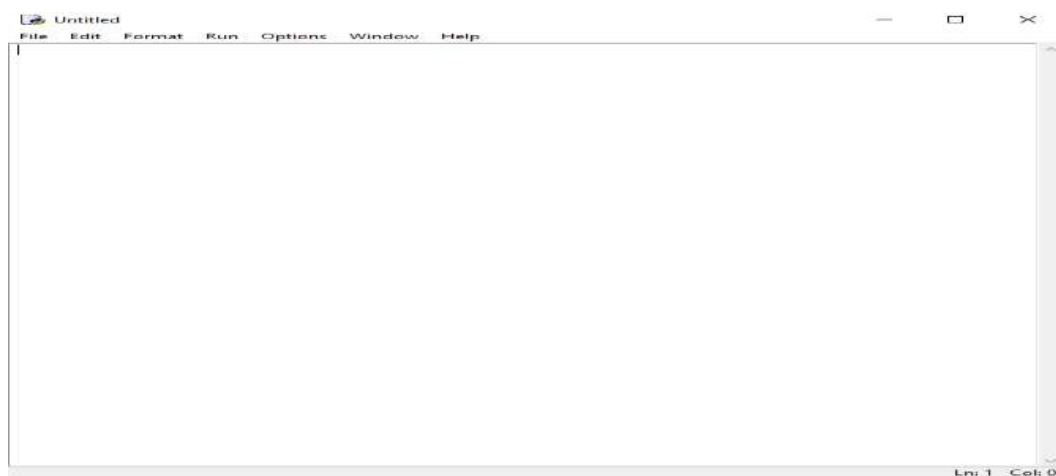
Μετά την εγκατάσταση θα ανοίξουμε το αρχείο IDLE (Python) και θα εμφανιστεί το παρακάτω παράθυρο:



Από την επιλογή File επιλέγουμε New File.



Έτσι θα έχουμε το κενό αρχείο, στο οποίο θα γράφουμε τον κώδικα.



Τώρα ήρθε η ώρα να μάθουμε τις εντολές και τον τρόπο που θα προγραμματίζουμε. Πρέπει να έχουμε πάντα στο μυαλό μας ότι ο υπολογιστής δεν έχει νοημοσύνη.

Άρα πρέπει οι εντολές που θα δώσουμε στον υπολογιστή να είναι σαφείς, ώστε να μην υπάρχει το ενδεχόμενο να φτάσουμε σε αδιέξοδο. Αν ο υπολογιστής δεν μπορεί να εκτελέσει κάτι, εμφανίζει μήνυμα λάθους.

Πάντα να θυμόμαστε ότι ο υπολογιστής δεν κάνει ποτέ λάθος. Λάθος κάνει ο άνθρωπος που τον προγραμματίζει.....!

ΚΕΦ 2. ΕΝΤΟΛΕΣ ΕΙΣΟΔΟΥ – ΕΞΟΔΟΥ

2.1 ΕΝΤΟΛΗ print

Με την εντολή print ο υπολογιστής εμφανίζει στην οθόνη ο,τιδήποτε ο προγραμματιστής επιθυμεί.

Γράφουμε πάντα με μικρά γράμματα:

```
print("Hello, Python")
```

Ο υπολογιστής θα εμφανίσει ό,τι χαρακτήρα υπάρχει μεταξύ των " " .

Δηλαδή:

```
Hello, Python
```

Αν μετά την εντολή print δεν υπάρχουν " " αλλά κάποια μεταβλητή, ο υπολογιστής εμφανίζει το περιεχόμενο της τιμής της μεταβλητής.

Για παράδειγμα:

```
x=2018
```

```
print(x)
```

Ο Υπολογιστής θα εμφανίσει :

```
2018
```

Μια μεταβλητή μπορεί να είναι αριθμός ή συνδυασμός χαρακτήρων(κείμενο).

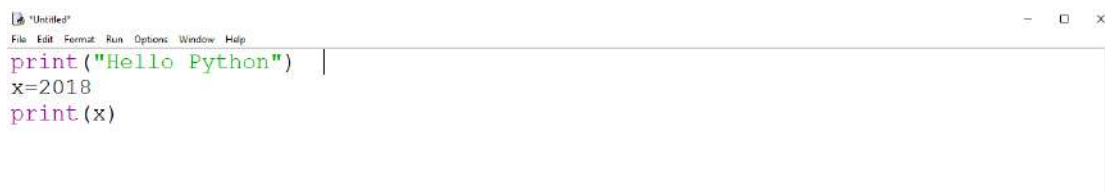
Συγκεκριμένα:

Ακέραιη μεταβλητή : int

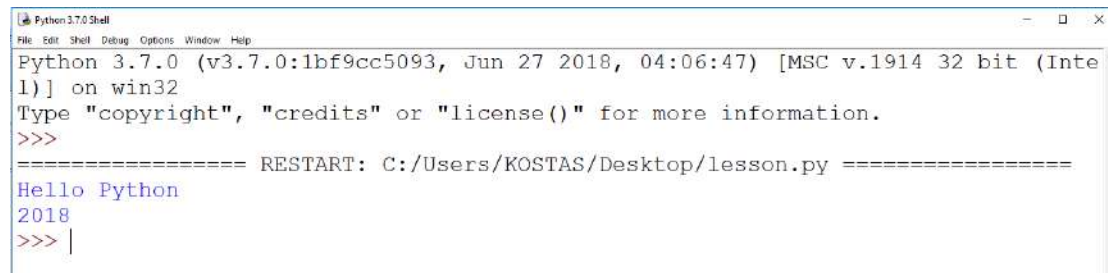
Πραγματική μεταβλητή : float

Κείμενο : string

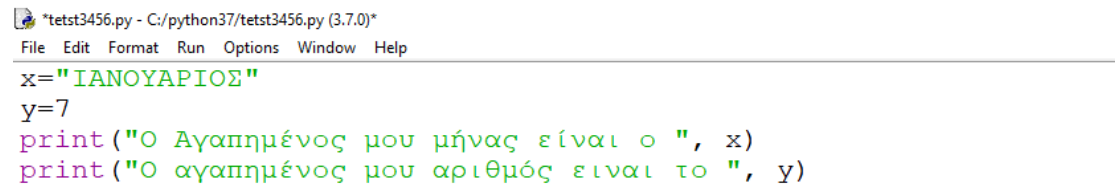
Ας δούμε τώρα πώς θα γίνει αυτό αναλυτικά με το αρχείο που δημιουργήσαμε:



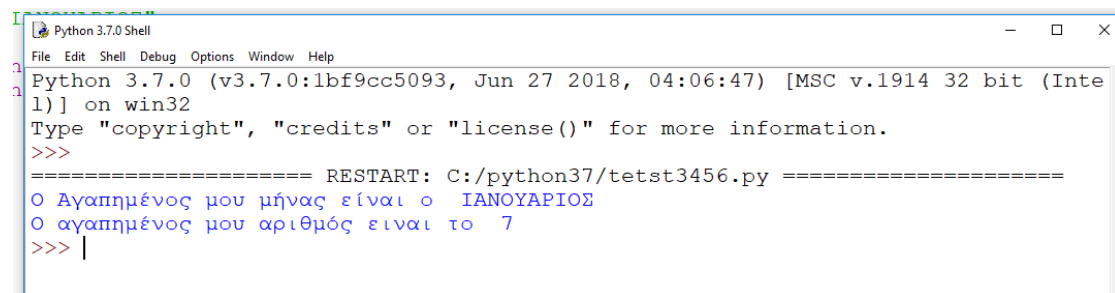
```
File Edit Format Run Options Window Help
print("Hello Python")
x=2018
print(x)
```

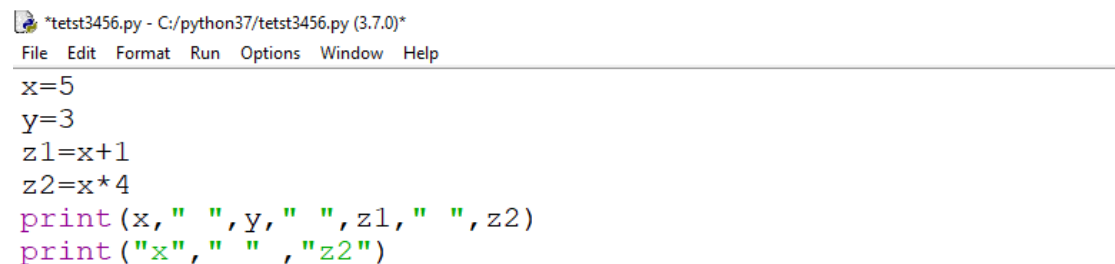
```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/lesson.py =====
Hello Python
2018
>>> |
```

Παράδειγμα:


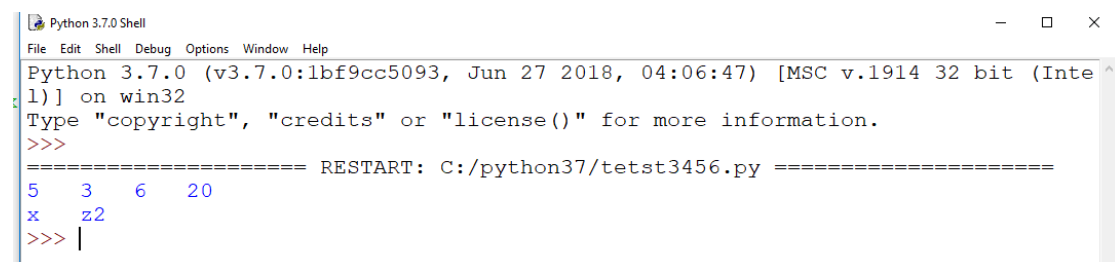
```
*tetst3456.py - C:/python37/tetst3456.py (3.7.0)*
File Edit Format Run Options Window Help
x="ΙΑΝΟΥΑΡΙΟΣ"
y=7
print("Ο Αγαπημένος μου μήνας είναι ο ", x)
print("Ο αγαπημένος μου αριθμός είναι το ", y)
```

ΕΚΤΕΛΕΣΗ:


```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/python37/tetst3456.py =====
Ο Αγαπημένος μου μήνας είναι ο ΙΑΝΟΥΑΡΙΟΣ
Ο αγαπημένος μου αριθμός είναι το 7
>>> |
```

Παράδειγμα:


```
*tetst3456.py - C:/python37/tetst3456.py (3.7.0)*
File Edit Format Run Options Window Help
x=5
y=3
z1=x+1
z2=x*4
print(x, " ", y, " ", z1, " ", z2)
print("x", " ", "z2")
```

ΕΚΤΕΛΕΣΗ:


```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/python37/tetst3456.py =====
5 3 6 20
x z2
>>> |
```

Από τα παραπάνω παραδείγματα μπορούμε να καταλάβουμε πώς λειτουργεί η εντολή print.

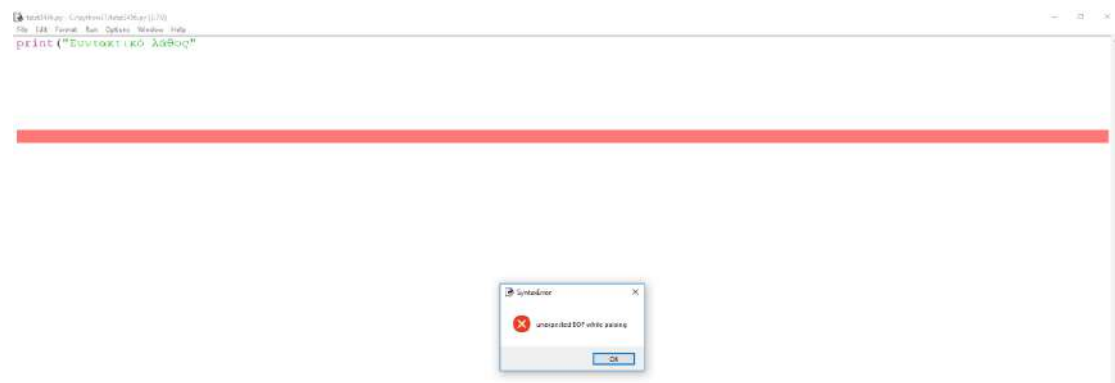
Τώρα θα δούμε τι θα συμβεί, αν κάνουμε κάποιο συντακτικό λάθος.

Στον παρακάτω κώδικα ξεχάσαμε να κλείσουμε την παρένθεση.



```
*tetst3456.py - C:/python37/tetst3456.py (3.7.0)*
File Edit Format Run Options Window Help
print ("Συντακτικό λάθος"
```

Εκτέλεση



```
*tetst3456.py - C:/python37/tetst3456.py (3.7.0)*
File Edit Format Run Options Window Help
print ("Συντακτικό λάθος"
```

Συντακτικό

υποσυνολικό EOF είναι αναίτιο

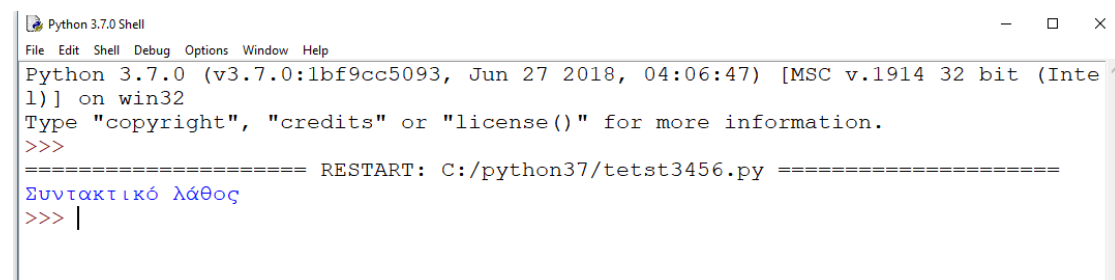
OK

Εμφανίζει μήνυμα λάθους. Διορθώνω το λάθος και το ξανατρέχω.



```
*tetst3456.py - C:/python37/tetst3456.py (3.7.0)*
File Edit Format Run Options Window Help
print ("Συντακτικό λάθος")
```

ΕΚΤΕΛΕΣΗ ΞΑΝΑ:



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/python37/tetst3456.py =====
Συντακτικό λάθος
>>> |
```

2.2 ΕΝΤΟΛΗ input

Η εντολή input είναι ίσως η πιο σημαντική εντολή, γιατί είναι αυτή που μας επιτρέπει να επικοινωνούμε άμεσα με τον υπολογιστή.

Με την εντολή input μπορούμε να εισάγουμε δεδομένα στον υπολογιστή. Η σύνταξη της εντολής input είναι κάπως ιδιότυπη. Δηλαδή:

```
x=input("κείμενο")
```

Παράδειγμα:

```
x=input("Δώσε το όνομά σου")
```

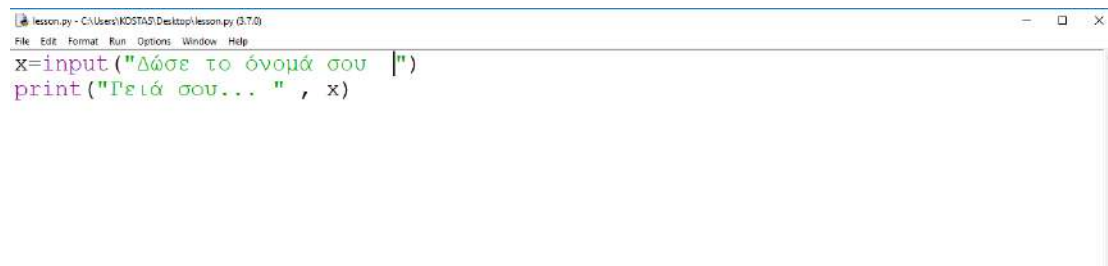
Όταν ο υπολογιστής εκτελέσει αυτήν την εντολή θα περιμένει από τον χρήστη να εισάγει από το πληκτρολόγιο μία τιμή, η οποία θα αποθηκευτεί στην μεταβλητή x ως χαρακτήρας.

Άλλες εκδοχές:

`x=int(input("Δώσε έναν ακέραιο αριθμό"))`, για ακέραια τιμή που θα εισάγει ο χρήστης. Ο υπολογιστής θα δεχτεί μόνο ακέραιο αριθμό.

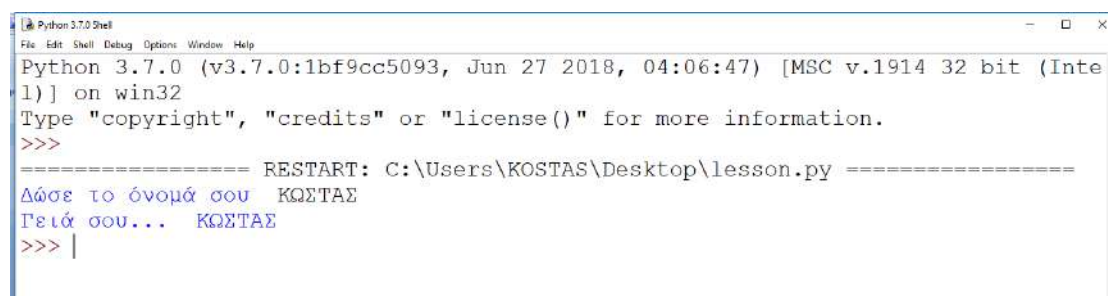
`x=float(input("Δώσε έναν πραγματικό αριθμό"))`, για πραγματική τιμή που θα εισάγει ο χρήστης. Ο υπολογιστής θα δεχτεί μόνο πραγματικό αριθμό.

Πρόγραμμα 1: Ο χρήστης εισάγει συνδυασμό χαρακτήρων(κείμενο)

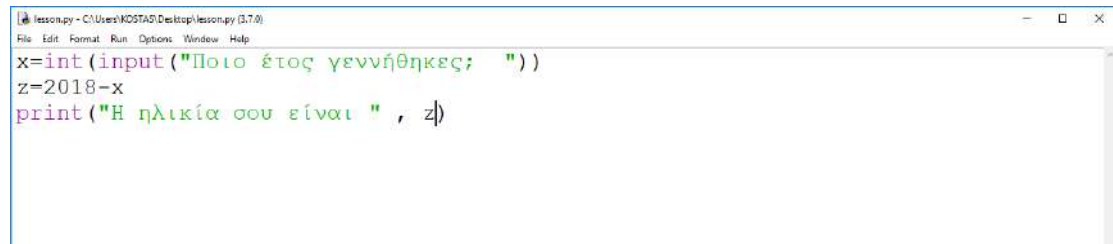


```
lesson.py - C:\Users\KOSTAS\Desktop\lesson.py (3.7.0)
File Edit Format Run Options Window Help
x=input("Δώσε το όνομά σου ")
print("Γειά σου... ", x)
```

Εκτέλεση:



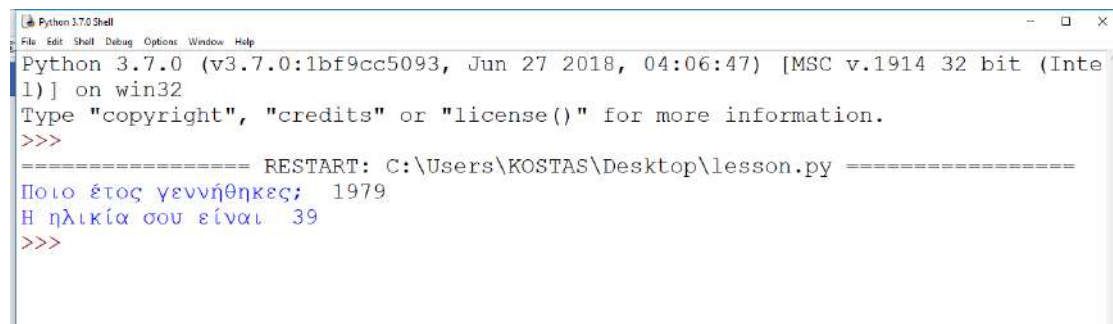
```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\KOSTAS\Desktop\lesson.py =====
Δώσε το όνομά σου ΚΩΣΤΑΣ
Γειά σου... ΚΩΣΤΑΣ
>>> |
```

Πρόγραμμα 2: Ο χρήστης πρέπει να εισάγει μόνο ακέραιο αριθμό.


```

lesson.py - C:\Users\KOSTAS\Desktop\lesson.py (3.7.0)
File Edit Format Run Options Window Help
x=int(input("Ποιο έτος γεννήθηκες; "))
z=2018-x
print("Η ηλικία σου είναι " , z)

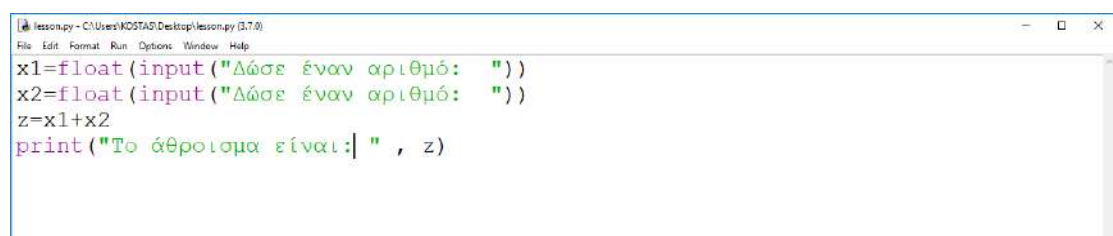
```

Εκτέλεση:


```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\KOSTAS\Desktop\lesson.py =====
Ποιο έτος γεννήθηκες; 1979
Η ηλικία σου είναι 39
>>>

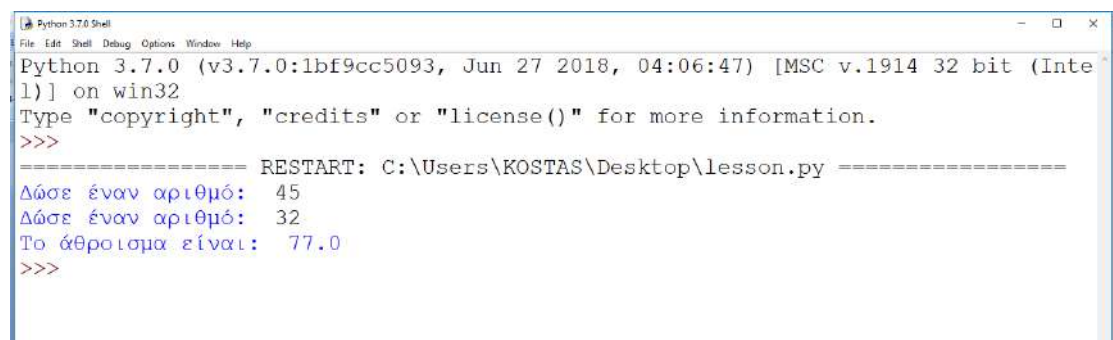
```

Παράδειγμα 3: Ο χρήστης πρέπει να εισάγει δύο πραγματικούς αριθμούς


```

lesson.py - C:\Users\KOSTAS\Desktop\lesson.py (3.7.0)
File Edit Format Run Options Window Help
x1=float(input("Δώσε έναν αριθμό: "))
x2=float(input("Δώσε έναν αριθμό: "))
z=x1+x2
print("Το άθροισμα είναι:| " , z)

```

Εκτέλεση:


```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\KOSTAS\Desktop\lesson.py =====
Δώσε έναν αριθμό: 45
Δώσε έναν αριθμό: 32
Το άθροισμα είναι: 77.0
>>>

```

Από τα παραπάνω παραδείγματα βλέπουμε ότι χρησιμοποιούνται ενδιάμεσες μεταβλητές, για να αποθηκευτούν πράξεις που εκτελούνται κατά την εκτέλεση του προγράμματος.

Σε ένα πρόγραμμα μπορούμε να χρησιμοποιήσουμε όσες μεταβλητές θέλουμε.

$z=x1+x2$ $z=2018-x$

2.2.1 Η εντολή input()

Με την εντολή input() έχουμε παύση της εκτέλεσης του προγράμματος. Για να συνεχιστεί η εκτέλεση του προγράμματος, πρέπει να πατηθεί το πλήκτρο ENTER.

Παράδειγμα:

```
'Untitled'  
File Edit Format Run Options Window Help  
name=input("Δώσε το όνομά σου ")  
input()  
print("Πάτησε το πλήκτρο ENTER για να σε καλωσορίσει η PYTHON....")  
print(name, " Καλωσήρθες στον κόσμο της Python")  
|
```

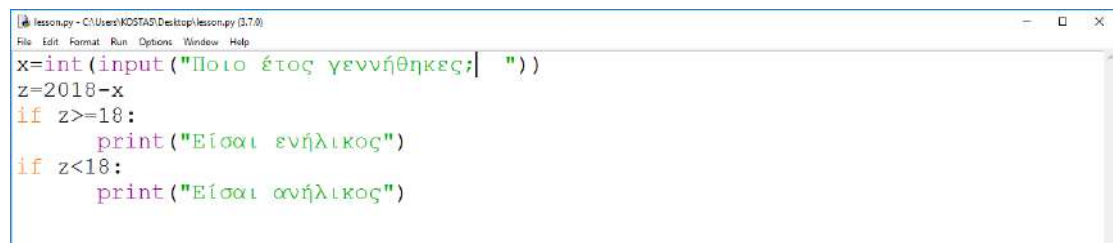
ΚΕΦ 3. ΕΝΤΟΛΕΣ ΕΠΙΛΟΓΗΣ

3.1.2. ΕΝΤΟΛΗ if

Με την χρήση της εντολής if ο υπολογιστής μπορεί να επιλέξει ποια συνθήκη θα ακολουθήσει, για να βγάλει κάποιο συμπέρασμα. Πιο απλά επιλέγει ποιο μονοπάτι θα ακολουθήσει, για να καταλήξει σε κάποιο αποτέλεσμα.

Παράδειγμα 1. Να γίνει πρόγραμμα στην ρηθον όπου θα ζητείται από τον χρήστη να εισάγει το έτος γέννησής του και ο υπολογιστής θα εμφανίζει, αν ο χρήστης είναι ενήλικος ή ανήλικος.

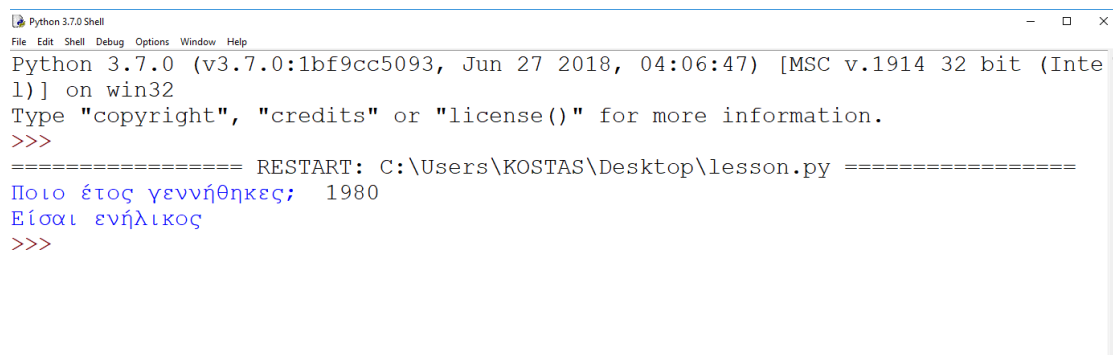
ΛΥΣΗ:



```

lesson.py - C:\Users\KOSTAS\Desktop\lesson.py (3.7.0)
File Edit Format Run Options Window Help
x=int(input("Ποιο έτος γεννήθηκες; "))
z=2018-x
if z>=18:
    print("Είσαι ενήλικος")
if z<18:
    print("Είσαι ανήλικος")
  
```

ΕΚΤΕΛΕΣΗ:

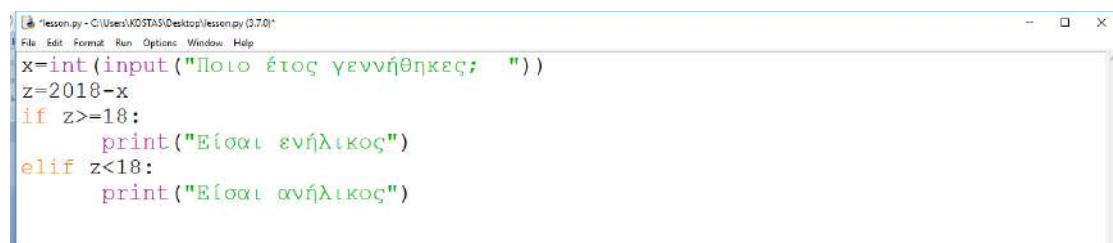


```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\KOSTAS\Desktop\lesson.py =====
Ποιο έτος γεννήθηκες; 1980
Είσαι ενήλικος
>>>
  
```

3.1.2. ΕΝΤΟΛΗ if elif

Λύση με elif (αλλιώς_αν)

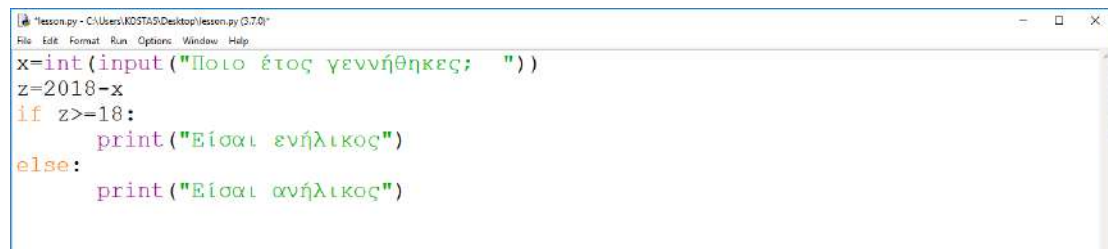


```

lesson.py - C:\Users\KOSTAS\Desktop\lesson.py (3.7.0)
File Edit Format Run Options Window Help
x=int(input("Ποιο έτος γεννήθηκες; "))
z=2018-x
if z>=18:
    print("Είσαι ενήλικος")
elif z<18:
    print("Είσαι ανήλικος")
  
```

3.1.3 ΕΝΤΟΛΗ if else

ΛΥΣΗ με else (αλλιώς)



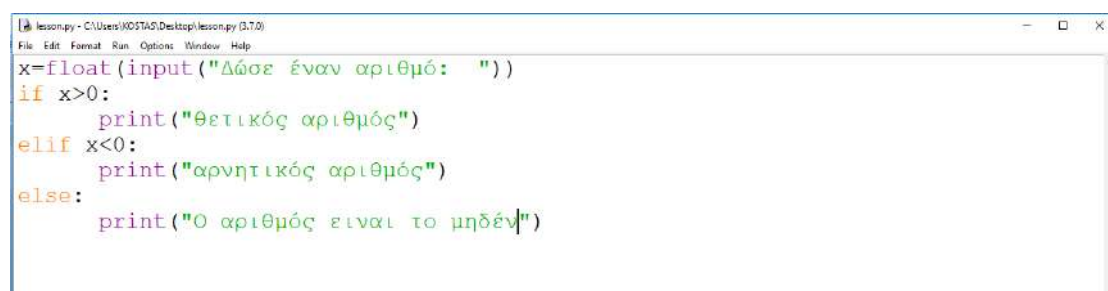
```

lesson.py - C:\Users\KOSTAS\Desktop\lesson.py (3.7.0)
File Edit Format Run Options Window Help
x=int(input("Ποιο έτος γεννήθηκες; "))
z=2018-x
if z>=18:
    print("Είσαι ενήλικος")
else:
    print("Είσαι ανήλικος")

```

Παράδειγμα 2. Να γίνει πρόγραμμα στην python όπου θα ζητείται από τον χρήστη να εισάγει έναν πραγματικό αριθμό και ο υπολογιστής θα εμφανίζει αν ο αριθμός είναι θετικός, αρνητικός ή μηδέν.

ΛΥΣΗ:

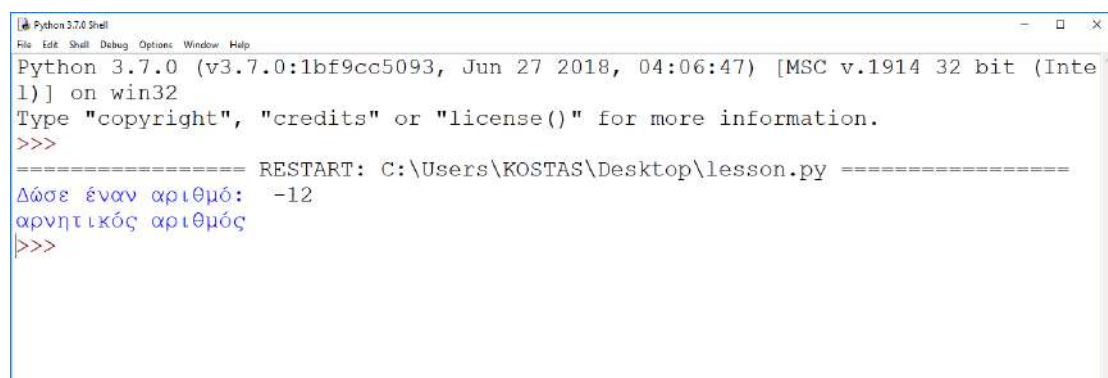


```

lesson.py - C:\Users\KOSTAS\Desktop\lesson.py (3.7.0)
File Edit Format Run Options Window Help
x=float(input("Δώσε έναν αριθμό: "))
if x>0:
    print("θετικός αριθμός")
elif x<0:
    print("αρνητικός αριθμός")
else:
    print("Ο αριθμός είναι το μηδέν")

```

ΕΚΤΕΛΕΣΗ:



```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\KOSTAS\Desktop\lesson.py =====
Δώσε έναν αριθμό: -12
αρνητικός αριθμός
>>>

```

Πρέπει να δείχνουμε μεγάλη προσοχή στη στοίχιση των εντολών και στη θέση της άνω κάτω τελείας.

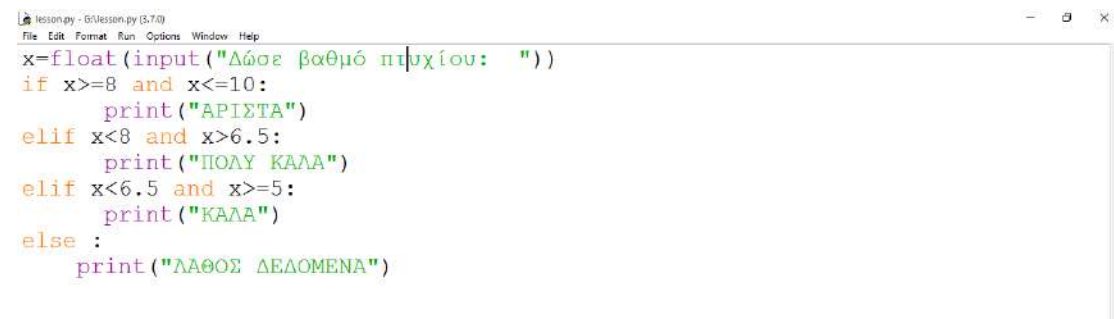
3.2. Οι τελεστές and και or

ΤΕΛΕΣΤΗΣ and

Παράδειγμα 1: Να γίνει πρόγραμμα στην ρηθον όπου θα ζητείται από τον χρήστη να εισάγει τον βαθμό ενός φοιτητή και ο υπολογιστής θα εμφανίζει τον χαρακτηρισμό της επίδοσης σύμφωνα με τον παρακάτω πίνακα:

ΒΑΘΜΟΣ ΠΤΥΧΙΟΥ	ΧΑΡΑΚΤΗΡΙΣΜΟΣ ΕΠΙΔΟΣΗΣ
9-10	ΑΡΙΣΤΑ
6,5-8	ΠΟΛΥ ΚΑΛΑ
5-6,5	ΚΑΛΑ

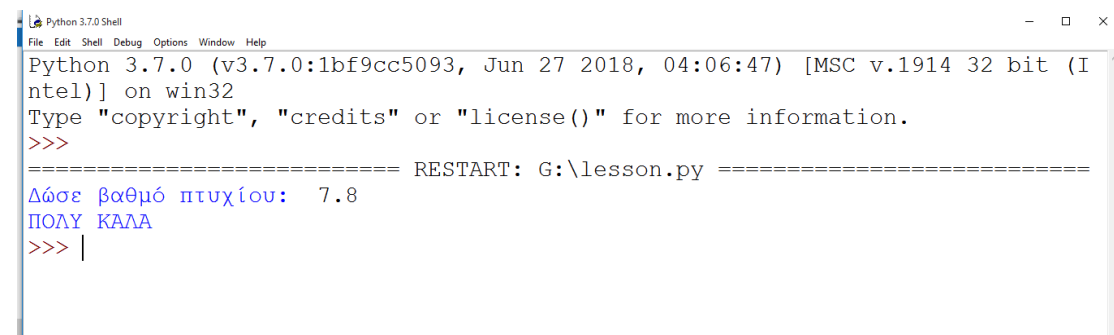
ΚΩΔΙΚΑΣ:



```

lesson.py - G:\lesson.py (3.7.0)
File Edit Format Run Options Window Help
x=float(input("Δώσε βαθμό πτυχίου: "))
if x>=8 and x<=10:
    print("ΑΡΙΣΤΑ")
elif x<8 and x>6.5:
    print("ΠΟΛΥ ΚΑΛΑ")
elif x<6.5 and x>=5:
    print("ΚΑΛΑ")
else :
    print("ΛΑΘΟΣ ΔΕΔΟΜΕΝΑ")
  
```

ΕΚΤΕΛΕΣΗ:



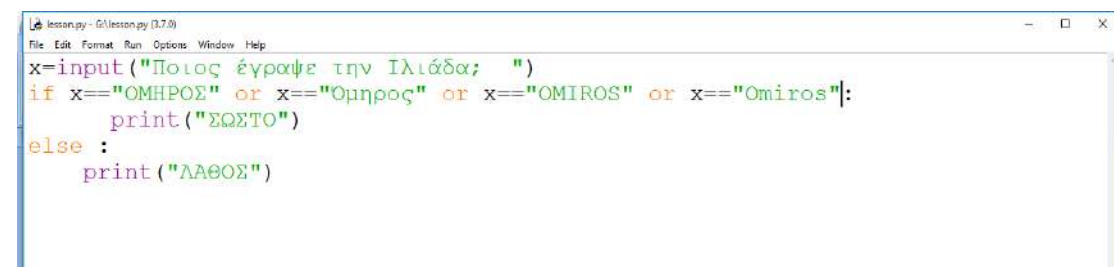
```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: G:\lesson.py =====
Δώσε βαθμό πτυχίου: 7.8
ΠΟΛΥ ΚΑΛΑ
>>> |
  
```

ΤΕΛΕΣΤΗΣ Or

Παράδειγμα 2: Να γίνει πρόγραμμα στην ρηθον όπου θα ζητείται από τον χρήστη να απαντήσει σε μία ερώτηση και ο υπολογιστής θα εμφανίζει μήνυμα ΣΩΣΤΟ ή ΛΑΘΟΣ.

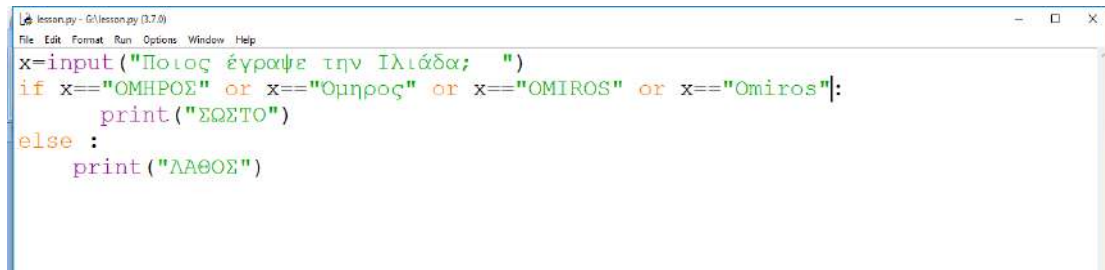
ΚΩΔΙΚΑΣ:



```

lesson.py - G:\lesson.py (3.7.0)
File Edit Format Run Options Window Help
x=input("Ποιος έγραψε την Ιλιάδα; ")
if x=="ΟΜΗΡΟΣ" or x=="Ομηρος" or x=="OMIROS" or x=="Omiros":
    print("ΣΩΣΤΟ")
else :
    print("ΛΑΘΟΣ")
  
```

ΕΚΤΕΛΕΣΗ:



```

lesson.py - G:\lesson.py (3.7.0)
File Edit Format Run Options Window Help
x=input("Ποιος έγραψε την Ιλιάδα; ")
if x=="ΟΜΗΡΟΣ" or x=="Ομηρος" or x=="OMIROS" or x=="Omiros":
    print("ΣΩΣΤΟ")
else :
    print("ΛΑΘΟΣ")

```

ΤΕΛΕΣΤΕΣ ΣΤΗΝ ΡΥΤΗΟΝ

=	Ίσον , παράδειγμα x=5
+	Πρόσθεση, παράδειγμα x=y+5
-	Αφαίρεση, παράδειγμα x=y-5
/	Διαίρεση, παράδειγμα x=y/5
*	Γινόμενο, παράδειγμα x=y*5
==	Ισότητα σε συνθήκη, if x==5, while x==5
!=	Ανισότητα σε συνθήκη, if x!=5, while x!=5
>	Μεγαλύτερο, παράδειγμα x>5
<	Μικρότερο, παράδειγμα x<5
>=	Μεγαλύτερο ή ίσο, παράδειγμα x>=5
<=	Μικρότερο ή ίσο , παράδειγμα x<=5
%	Mod, παράδειγμα x%2=
//	Div, παράδειγμα x//2=
**	Εκθέτης, παράδειγμα x**2
and	και, παράδειγμα x=5 and y=3
or	ή, παράδειγμα x=5 ή x=4

ΚΕΦ 4. ΕΝΤΟΛΕΣ ΔΟΜΗΣ ΕΠΑΝΑΛΗΨΗΣ

4.1. ΕΝΤΟΛΗ while

Με την εντολή while ο υπολογιστής κάνει συνεχόμενες επαναλήψεις μέχρι μια συνθήκη να σταματήσει να ισχύει. Όταν συμβεί αυτό, θα σταματήσουν οι επαναλήψεις.

Με τη χρήση της εντολής while μπορεί να λυθεί κάθε πρόβλημα που χρειάζεται επαναλήψεις, είτε πεπερασμένο αριθμό επαναλήψεων, είτε αόριστο αριθμό, όπου ο χρήστης θα καθορίσει με το πάτημα πλήκτρου πότε θα σταματήσει η επανάληψη.

1. Πεπερασμένος αριθμός επαναλήψεων:

```
k=1
While x<=100:
    print("PYTHON")
    k=k+1
# Η λέξη PYTHON θα εμφανιστεί 100 φορές
```

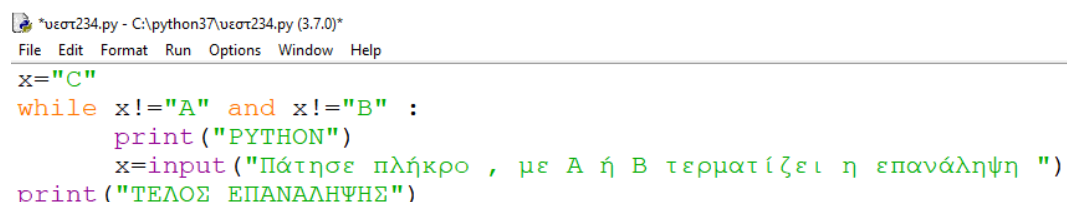
2. Αόριστος αριθμός επαναλήψεων

```
x=1
while x!=0
x=int("Δώσε αριθμό")
print("PYTHON")
```

#Αν εκτελεστεί ο παραπάνω κώδικας, κάθε φορά που ο χρήστης εισάγει τιμή διάφορη του μηδενός, θα εμφανίζεται η λέξη PYTHON. Αν εισάγει την τιμή μηδέν, θα σταματήσει η επανάληψη.

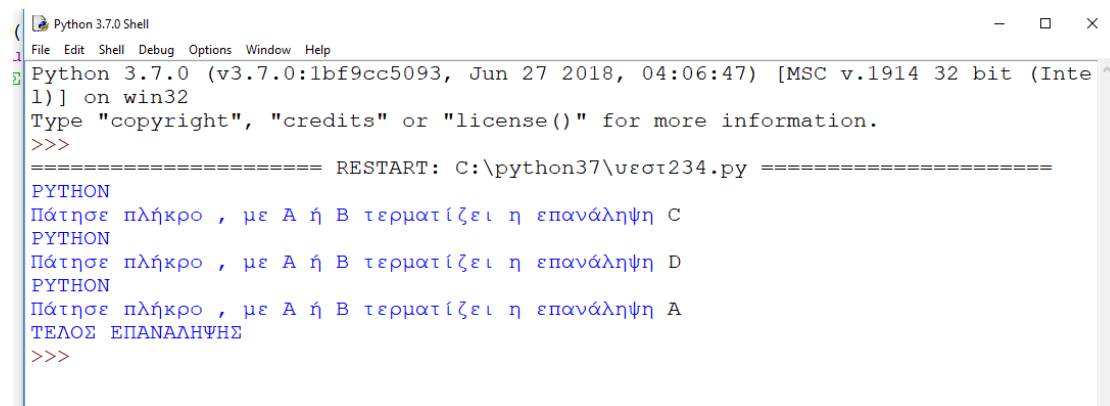
Πρέπει να δείχνουμε μεγάλη προσοχή στη στοίχιση των εντολών και στη θέση της άνω κάτω τελείας.

Παράδειγμα 1: Να γίνει πρόγραμμα στην python όπου ο υπολογιστής θα εμφανίζει επαναληπτικά την φράση «Μου αρέσει πολύ η Python», μέχρι να πατηθεί το πλήκτρο A ή B.



```
~\εεστ234.py - C:\python37\εεστ234.py (3.7.0)*
File Edit Format Run Options Window Help
x="C"
while x!="A" and x!="B" :
    print("PYTHON")
    x=input("Πάτησε πλήκρο , με A ή B τερματίζει η επανάληψη ")
print("ΤΕΛΟΣ ΕΠΑΝΑΛΗΨΗΣ")
```

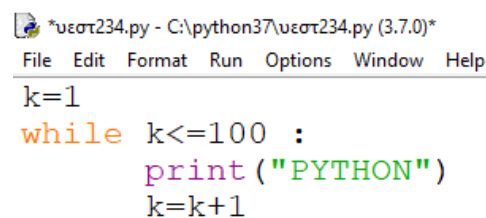
ΕΚΤΕΛΕΣΗ:



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\python37\υεστ234.py =====
PYTHON
Πάτησε πλήκρο , με A ή B τερματίζει η επανάληψη C
PYTHON
Πάτησε πλήκρο , με A ή B τερματίζει η επανάληψη D
PYTHON
Πάτησε πλήκρο , με A ή B τερματίζει η επανάληψη A
ΤΕΛΟΣ ΕΠΑΝΑΛΗΨΗΣ
>>>
```

Παράδειγμα 2: Να γίνει πρόγραμμα με τη χρήση `while` στην `python` όπου ο υπολογιστής θα εμφανίζει τη λέξη `PYTHON` 100 φορές.

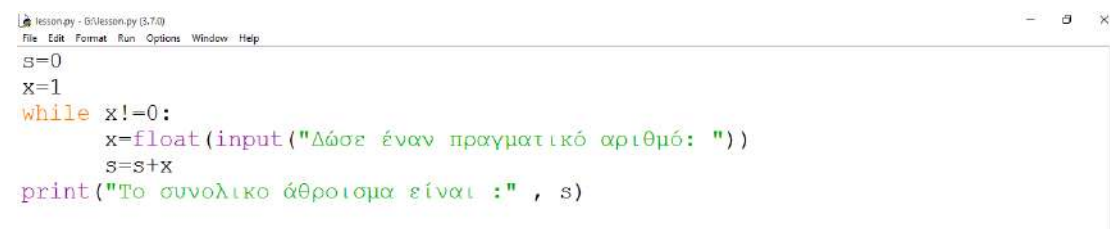
ΚΩΔΙΚΑΣ:



```
*υεστ234.py - C:\python37\υεστ234.py (3.7.0)*
File Edit Format Run Options Window Help
k=1
while k<=100 :
    print("PYTHON")
    k=k+1
```

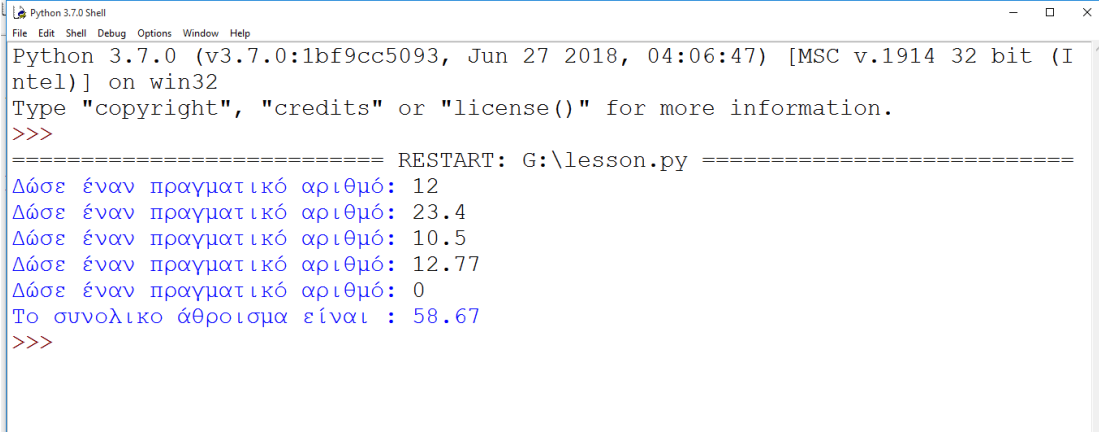
Παράδειγμα 3: Να γίνει πρόγραμμα στην `python` όπου ο χρήστης θα εισάγει συνεχόμενα πραγματικούς αριθμούς. Όταν ο χρήστης θα θέλει να σταματήσει την εισαγωγή, θα πατήσει το πλήκτρο και ο υπολογιστής θα εμφανίζει το συνολικό άθροισμα.

ΚΩΔΙΚΑΣ:



```
lesson.py - G:\lesson.py (3.7.0)
File Edit Format Run Options Window Help
s=0
x=1
while x!=0:
    x=float(input("Δώσε έναν πραγματικό αριθμό: "))
    s=s+x
print("Το συνολικό άθροισμα είναι :", s)
```

ΕΚΤΕΛΕΣΗ:



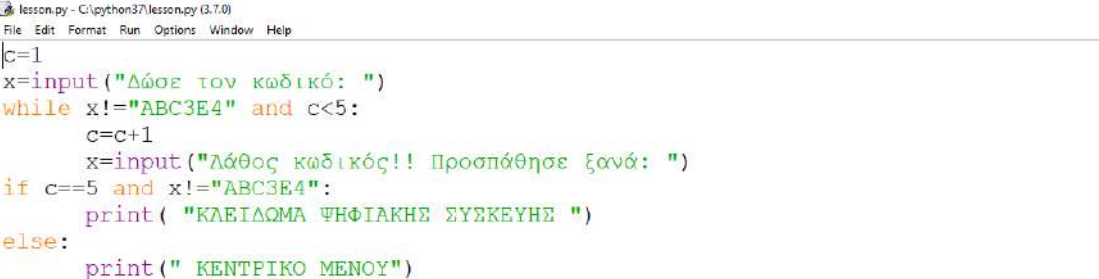
```

Python 3.7.0 Shell
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: G:\lesson.py =====
Δώσε έναν πραγματικό αριθμό: 12
Δώσε έναν πραγματικό αριθμό: 23.4
Δώσε έναν πραγματικό αριθμό: 10.5
Δώσε έναν πραγματικό αριθμό: 12.77
Δώσε έναν πραγματικό αριθμό: 0
Το συνολικο άθροισμα είναι : 58.67
>>>

```

ΠΑΡΑΔΕΙΓΜΑ 4: Σε μια ψηφιακή συσκευή, για να μπορέσει ο χρήστης να μπει στο κεντρικό μενού, πρέπει να δώσει τον κωδικό ACB3E4. Αν δώσει 5 φορές λάθος κωδικό, η ψηφιακή συσκευή κλειδώνει. Να γίνει πρόγραμμα στην python που να υλοποιεί την παραπάνω διαδικασία.

ΚΩΔΙΚΑΣ:

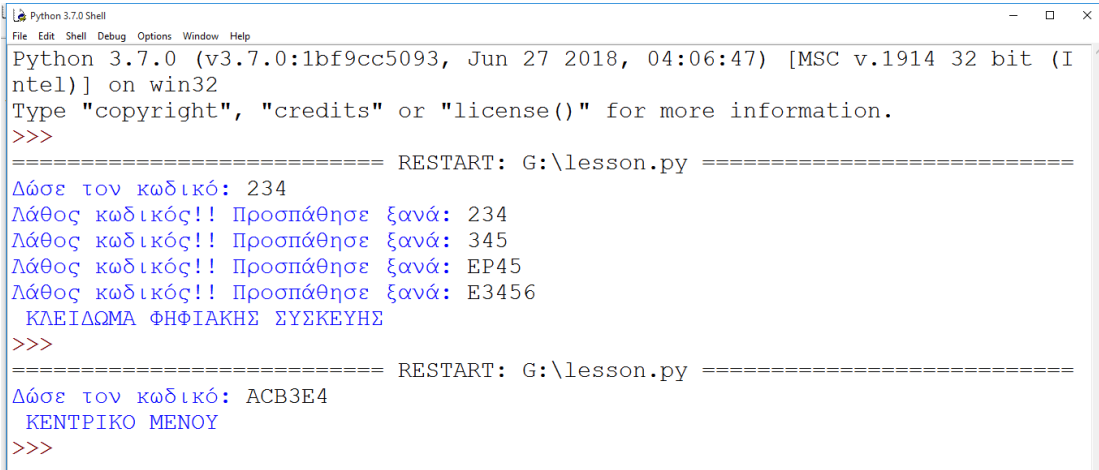


```

lesson.py - C:\python37\lesson.py (3.7.0)
File Edit Format Run Options Window Help
c=1
x=input("Δώσε τον κωδικό: ")
while x!="ACB3E4" and c<5:
    c=c+1
    x=input("Λάθος κωδικός!! Προσπάθησε ξανά: ")
if c==5 and x!="ACB3E4":
    print(" ΚΛΕΙΔΩΜΑ ΨΗΦΙΑΚΗΣ ΣΥΣΚΕΥΗΣ ")
else:
    print(" ΚΕΝΤΡΙΚΟ ΜΕΝΟΥ")

```

ΕΚΤΕΛΕΣΗ:



```

Python 3.7.0 Shell
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: G:\lesson.py =====
Δώσε τον κωδικό: 234
Λάθος κωδικός!! Προσπάθησε ξανά: 234
Λάθος κωδικός!! Προσπάθησε ξανά: 345
Λάθος κωδικός!! Προσπάθησε ξανά: EP45
Λάθος κωδικός!! Προσπάθησε ξανά: E3456
ΚΛΕΙΔΩΜΑ ΨΗΦΙΑΚΗΣ ΣΥΣΚΕΥΗΣ
>>>
===== RESTART: G:\lesson.py =====
Δώσε τον κωδικό: ACB3E4
ΚΕΝΤΡΙΚΟ ΜΕΝΟΥ
>>>

```

4.2. ΕΝΤΟΛΗ for

Με την εντολή for ο υπολογιστής κάνει πεπερασμένο αριθμό επαναλήψεων που ορίζει ο χρήστης με αντίστοιχο βήμα. Η εντολή for δεν χρησιμοποιείται για αόριστο αριθμό επαναλήψεων. Συνήθως χρησιμοποιείται μαζί με λίστες (Πίνακες).

ΣΥΝΤΑΞΗ ΕΝΤΟΛΗΣ:

for i in range (αρχή, τέλος, βήμα):

ΕΝΤΟΛΕΣ

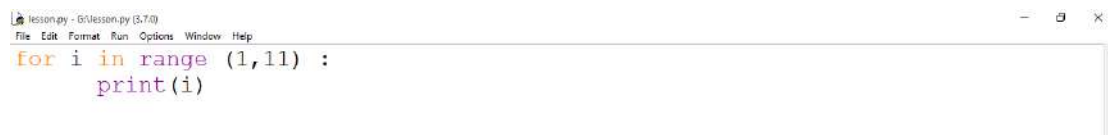
Πρέπει να δείχνουμε μεγάλη προσοχή στη στοίχιση των εντολών και στη θέση της άνω κάτω τελείας.

```
For i in range(2,100,2) :
```

Σημαίνει ότι η μεταβλητή i σε κάθε επανάληψη θα αυξάνει κατά δύο. Όταν η μεταβλητή i γίνει ίση ή μεγαλύτερη το 100 θα σταματήσει η επανάληψη.

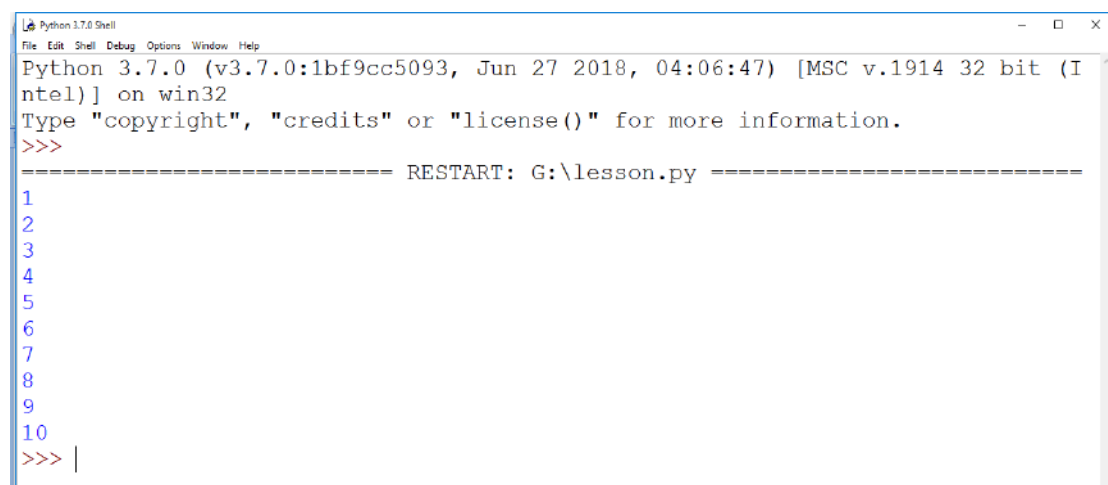
Παράδειγμα 1: Να γίνει πρόγραμμα στην python όπου θα εμφανίζονται όλοι οι ακέραιοι αριθμοί από το 1 έως και το 10.

ΚΩΔΙΚΑΣ:



```
lesson.py - G:\lesson.py (3.7.0)
File Edit Format Run Options Window Help
for i in range (1,11) :
    print(i)
```

ΕΚΤΕΛΕΣΗ:



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (I
ntel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
----- RESTART: G:\lesson.py -----
1
2
3
4
5
6
7
8
9
10
>>> |
```

Παράδειγμα 2: Να γίνει πρόγραμμα στην python όπου θα εμφανίζονται όλοι οι άρτιοι ακέραιοι αριθμοί από το 2 έως και το 20.

ΚΩΔΙΚΑΣ:

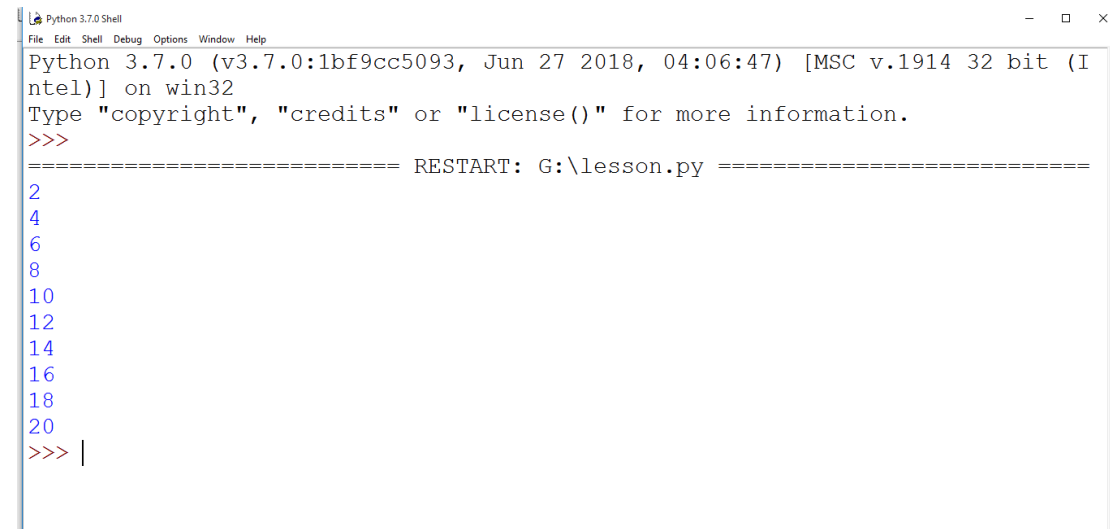


```

lesson.py - G:\lesson.py (3.7.0)
File Edit Format Run Options Window Help
for i in range (2,21,2) :
    print(i)

```

ΕΚΤΕΛΕΣΗ:



```

Python 3.7.0 Shell
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: G:\lesson.py =====
2
4
6
8
10
12
14
16
18
20
>>> |

```

Παράδειγμα 3: Να γίνει πρόγραμμα στην python όπου θα εμφανίζεται η λέξη python 20 φορές.

ΛΥΣΗ:




```

lesson.py - G:\lesson.py (3.7.0)
File Edit Format Run Options Window Help
for i in range (1,21) :
    print("python")

```

Παράδειγμα 4: Να γίνει πρόγραμμα στην python όπου θα υπολογίζεται και μετά θα εμφανίζεται το αποτέλεσμα του αθροίσματος 1+3+5+7+....+99.

ΛΥΣΗ:

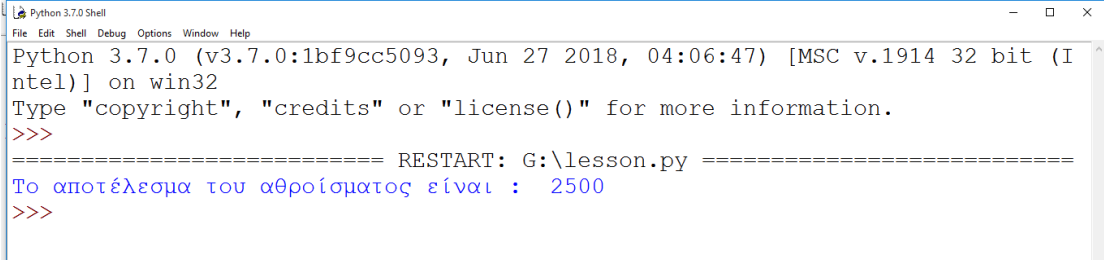


```

lesson.py - G:\lesson.py (3.7.0)
File Edit Format Run Options Window Help
s=0
for i in range (1,100,2) :
    s=s+i
print("Το αποτέλεσμα του αθροίσματος είναι : " , s)

```

ΕΚΤΕΛΕΣΗ:



```
Python 3.7.0 Shell
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: G:\lesson.py =====
Το αποτέλεσμα του αθροίσματος είναι : 2500
>>>
```

4.2.1. ΤΕΛΕΣΤΕΣ div, mod

Ο τελεστής div δίνει το ηλίκιο μια διαίρεσης.

Ο τελεστής mod δίνει το υπόλοιπο μιας διαίρεσης.

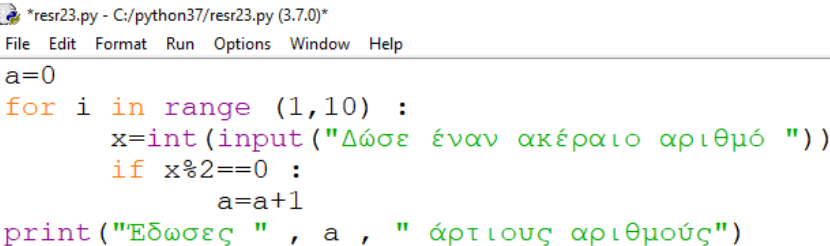
Στη python ο τελεστής div συμβολίζεται με //, ενώ ο τελεστής mod με %

$11//2=5$

$11\%2=1$

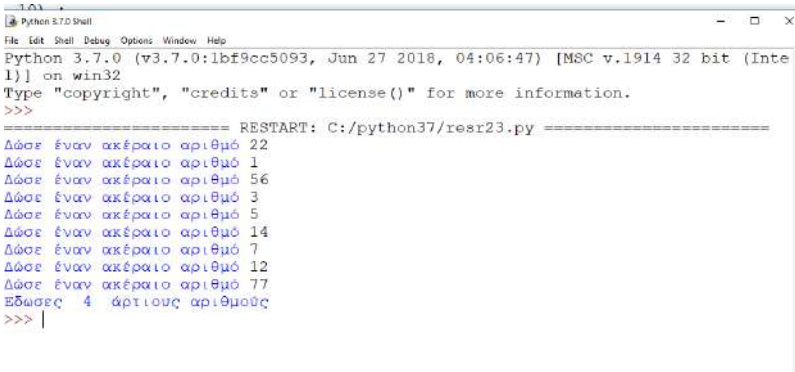
Παράδειγμα 1: Να γίνει πρόγραμμα σε Python όπου ο χρήστης θα εισάγει 9 αριθμούς και ο υπολογιστής θα εμφανίζει πόσοι από αυτούς είναι άρτιοι.

ΚΩΔΙΚΑΣ:



```
*resr23.py - C:/python37/resr23.py (3.7.0)*
File Edit Format Run Options Window Help
a=0
for i in range (1,10) :
    x=int(input("Δώσε έναν ακέραιο αριθμό "))
    if x%2==0 :
        a=a+1
print("Έδωσες " , a , " άρτιους αριθμούς")
```

ΕΚΤΕΛΕΣΗ:



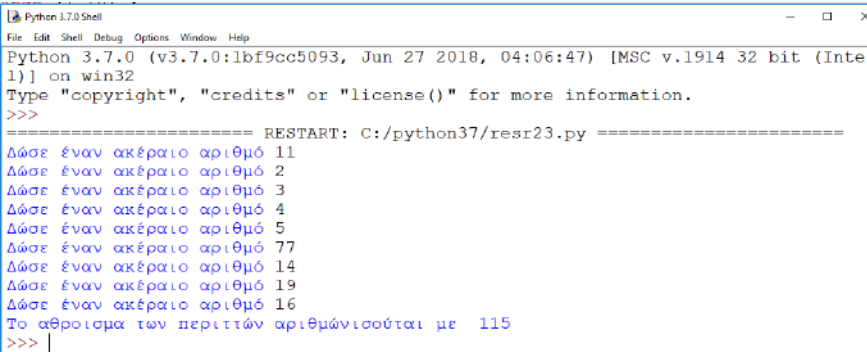
```
Python 3.7.0 Shell
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/python37/resr23.py =====
Δώσε έναν ακέραιο αριθμό 22
Δώσε έναν ακέραιο αριθμό 1
Δώσε έναν ακέραιο αριθμό 56
Δώσε έναν ακέραιο αριθμό 3
Δώσε έναν ακέραιο αριθμό 5
Δώσε έναν ακέραιο αριθμό 14
Δώσε έναν ακέραιο αριθμό 7
Δώσε έναν ακέραιο αριθμό 12
Δώσε έναν ακέραιο αριθμό 77
Έδωσες 4 άρτιους αριθμούς
>>>
```

Παράδειγμα 2: Να γίνει πρόγραμμα σε Python όπου ο χρήστης θα εισάγει 9 αριθμούς και ο υπολογιστής θα εμφανίζει το άθροισμα των περιττών αριθμών μόνο.

ΚΩΔΙΚΑΣ:

```
*resr23.py - C:/python37/resr23.py (3.7.0)*
File Edit Format Run Options Window Help
s=0
for i in range (1,10) :
    x=int(input("Δώσε έναν ακέραιο αριθμό "))
    if x%2==1 :
        s=s+x
print("Το αθροισμα των περιττών αριθμών ισούται με ", s)
```

ΕΚΤΕΛΕΣΗ:



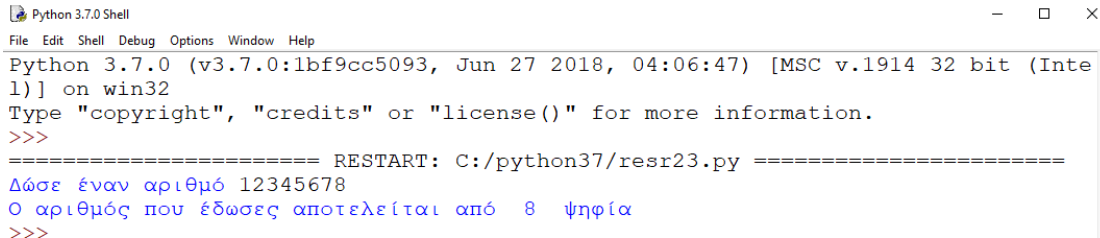
```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/python37/resr23.py =====
Δώσε έναν ακέραιο αριθμό 11
Δώσε έναν ακέραιο αριθμό 2
Δώσε έναν ακέραιο αριθμό 3
Δώσε έναν ακέραιο αριθμό 4
Δώσε έναν ακέραιο αριθμό 5
Δώσε έναν ακέραιο αριθμό 77
Δώσε έναν ακέραιο αριθμό 14
Δώσε έναν ακέραιο αριθμό 19
Δώσε έναν ακέραιο αριθμό 16
Το αθροισμα των περιττών αριθμών ισούται με 115
>>> |
```

Παράδειγμα 3. Να γίνει πρόγραμμα σε Python όπου ο χρήστης θα εισάγει έναν αριθμό και ο υπολογιστής θα εμφανίζει από πόσα ψηφία αποτελείται ο αριθμός αυτός.

ΚΩΔΙΚΑΣ:

```
*resr23.py - C:/python37/resr23.py (3.7.0)*
File Edit Format Run Options Window Help
x=int(input("Δώσε έναν αριθμό "))
n=1
while x//10!=0:
    p=x%10
    x=x//10
    n=n+1
print("Ο αριθμός που έδωσες αποτελείται από ", n , " ψηφία")
```

ΕΚΤΕΛΕΣΗ:



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/python37/resr23.py =====
Δώσε έναν αριθμό 12345678
Ο αριθμός που έδωσες αποτελείται από 8 ψηφία
>>>
```

Παράδειγμα 4: Ένα ATM δίνει χαρτονομίσματα των 50 και των 20 ευρώ. Ο πελάτης πρέπει να εισάγει πόσο στρόγγυλο μεγαλύτερο ή ίσο των 40 ευρώ. Να γίνει πρόγραμμα σε python που να υλοποιεί την παραπάνω διαδικασία.

ΚΩΔΙΚΑΣ:

```
atm.py - C:\Users\KOSTAS\Desktop\atm.py (3.7.0)
File Edit Format Run Options Window Help
print("ΠΡΟΓΡΑΜΜΑ ΑΤΜ")
x=int(input("Δώσε ποσό "))
while x%10!=0 or x<40:
    x=int(input("ΛΑΘΟΣ! Δώσε στρόγγυλο ποσό μεγαλύτερο ή ίσο του 40 "))
x50=x//50
y=x%50
if y%20!=0 :
    x50=x50-1
    y=y+50
x20=y//20
print("Πεντηντάρικα: ",x50, " Εικοσάρικα: " ,x20)
```

ΕΚΤΕΛΕΣΗ:

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (tags/v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
----- RESTART: C:\Users\KOSTAS\Desktop\atm.py -----
ΠΡΟΓΡΑΜΜΑ ΑΤΜ
Δώσε ποσό 130
Πεντηντάρικα: 1 Εικοσάρικα: 4
>>> |
```

ΚΕΦ. 5. ΤΥΧΑΙΟΙ ΑΡΙΘΜΟΙ

Η συνάρτηση `randrange` είναι υπεύθυνη για τη δημιουργία τυχαίων αριθμών.

Με τη συνάρτηση `randrange` μπορούμε να έχουμε τυχαίους αριθμούς τους οποίους παράγει ο υπολογιστής και μπορούμε να τους χρησιμοποιήσουμε για δική μας χρήση.

Αυτό μπορεί να γίνει με τη χρήση της εισαγωγής της βιβλιοθήκης `random`.

```
Import random
```

```
x=random.randrange(ακέραιος αριθμός 1,ακέραιος αριθμός 2)
```

Στην μεταβλητή `x` θα αποθηκευτεί ένας τυχαίος αριθμός που θα έχει τιμή:

Ακέραιος αριθμός $1 \leq x <$ ακέραιος αριθμός 2

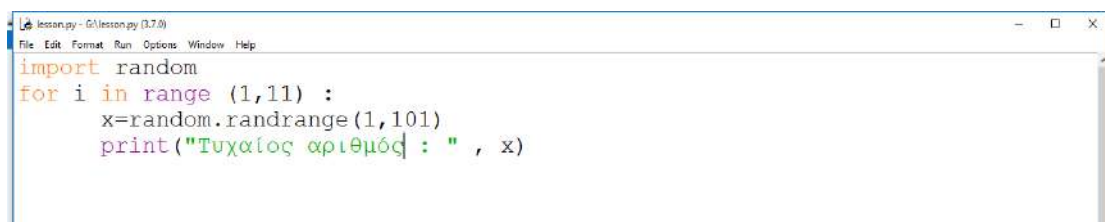
Δηλαδή:

```
x=random.randrange(0,1000)
```

Ο υπολογιστής θα αποθηκεύσει στην μεταβλητή `x` έναν ακέραιο αριθμό από το 1 μέχρι και το 999.

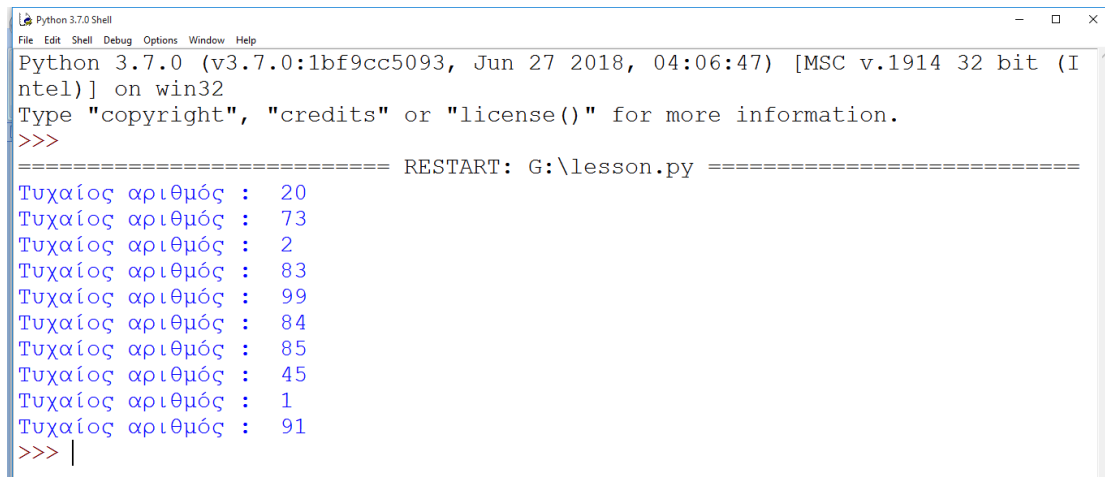
Παράδειγμα 1: Να γίνει πρόγραμμα στην `python` όπου θα εμφανίζονται 10 τυχαίοι αριθμοί που θα έχουν τιμή από 1 μέχρι και 100.

ΚΩΔΙΚΑΣ:



```
lesson.py - G:\lesson.py (3.7.0)
File Edit Format Run Options Window Help
import random
for i in range (1,11) :
    x=random.randrange(1,101)
    print("Τυχαίος αριθμός : " , x)
```

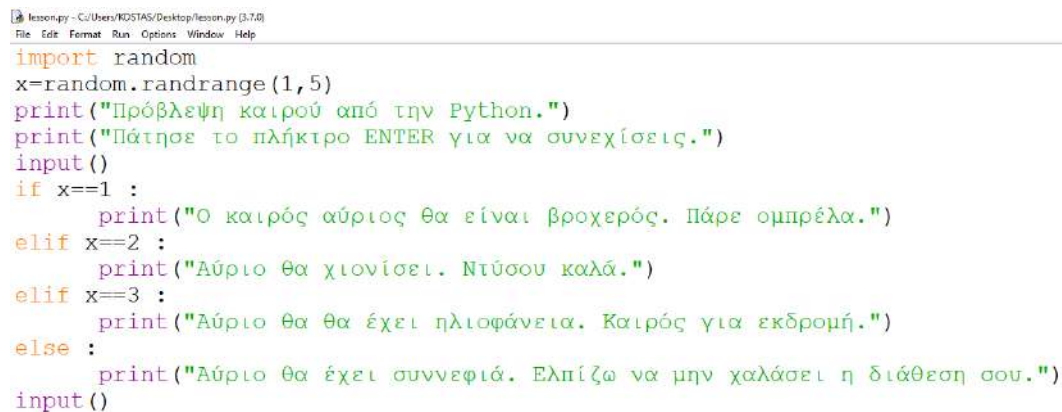
ΕΚΤΕΛΕΣΗ:



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: G:\lesson.py =====
Τυχαίος αριθμός : 20
Τυχαίος αριθμός : 73
Τυχαίος αριθμός : 2
Τυχαίος αριθμός : 83
Τυχαίος αριθμός : 99
Τυχαίος αριθμός : 84
Τυχαίος αριθμός : 85
Τυχαίος αριθμός : 45
Τυχαίος αριθμός : 1
Τυχαίος αριθμός : 91
>>> |
```

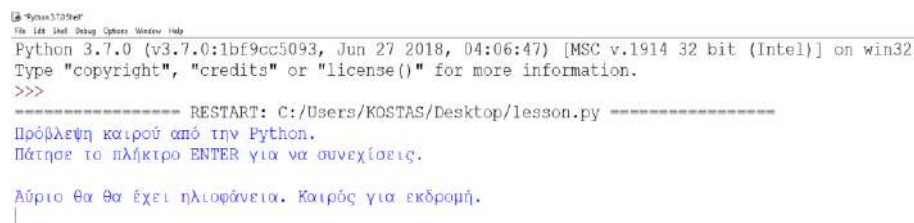
Παράδειγμα 2: Να γίνει πρόγραμμα όπου ο υπολογιστής θα κάνει τυχαία πρόβλεψη για τον αυριανό καιρό.

ΚΩΔΙΚΑΣ:



```
lesson.py - C:/Users/KOSTAS/Desktop/lesson.py (3.7.0)
File Edit Format Run Options Window Help
import random
x=random.randrange(1,5)
print("Πρόβλεψη καιρού από την Python.")
print("Πάτησε το πλήκτρο ENTER για να συνεχίσεις.")
input()
if x==1 :
    print("Ο καιρός αύριος θα είναι βροχερός. Πάρε ομπρέλα.")
elif x==2 :
    print("Αύριο θα χιονίσει. Ντύσου καλά.")
elif x==3 :
    print("Αύριο θα έχει ηλιοφάνεια. Καιρός για εκδρομή.")
else :
    print("Αύριο θα έχει συννεφιά. Ελπίζω να μην χαλάσει η διάθεση σου.")
input()
```

ΕΚΤΕΛΕΣΗ:

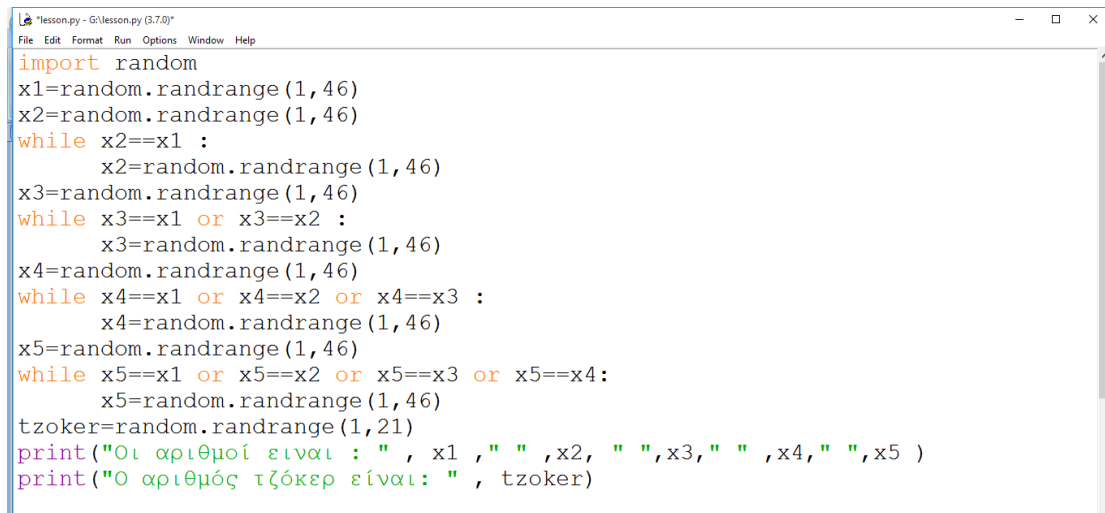


```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/lesson.py =====
Πρόβλεψη καιρού από την Python.
Πάτησε το πλήκτρο ENTER για να συνεχίσεις.

Αύριο θα έχει ηλιοφάνεια. Καιρός για εκδρομή.
```

Παράδειγμα 3: Να γίνει πρόγραμμα στην python που θα εμφανίζει τους πιθανούς αριθμούς της κλήρωσης Τζόκερ.

ΚΩΔΙΚΑΣ:

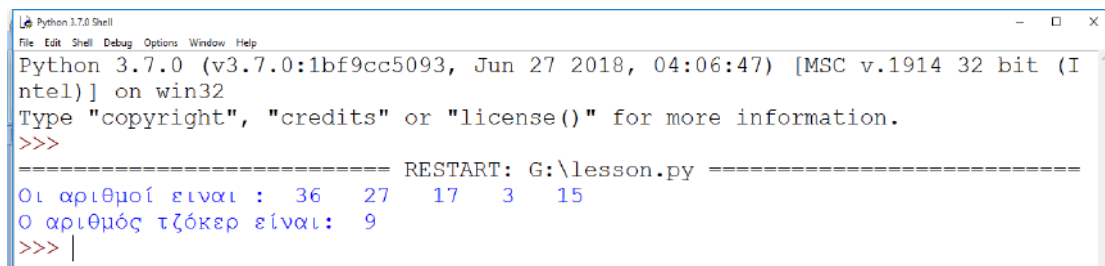


```

import random
x1=random.randrange(1,46)
x2=random.randrange(1,46)
while x2==x1 :
    x2=random.randrange(1,46)
x3=random.randrange(1,46)
while x3==x1 or x3==x2 :
    x3=random.randrange(1,46)
x4=random.randrange(1,46)
while x4==x1 or x4==x2 or x4==x3 :
    x4=random.randrange(1,46)
x5=random.randrange(1,46)
while x5==x1 or x5==x2 or x5==x3 or x5==x4:
    x5=random.randrange(1,46)
tzoker=random.randrange(1,21)
print("Οι αριθμοί είναι : " , x1 , " " ,x2, " " ,x3," " ,x4," " ,x5 )
print("Ο αριθμός τζόκερ είναι: " , tzoker)

```

ΕΚΤΕΛΕΣΗ:



```

Python 3.7.0 Shell
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: G:\lesson.py =====
Οι αριθμοί είναι : 36 27 17 3 15
Ο αριθμός τζόκερ είναι: 9
>>> |

```

Παράδειγμα 4: Να γίνει πρόγραμμα σε ρυθμό που θα κάνει τα παρακάτω: Ο υπολογιστής θα επιλέγει έναν τυχαίο αριθμό από 1 έως και το 50. Ο χρήστης θα έχει 5 προσπάθειες να μαντέψει τον αριθμό. Αν ο χρήστης μαντεύει λάθος αριθμό, ο υπολογιστής θα ενημερώνει αν ο αριθμός που αναζητείται είναι μικρότερος ή μεγαλύτερος. Αν ο χρήστης μαντέψει σωστά τον αριθμό, θα εμφανίζεται μήνυμα «Μπράβο! Βρήκες τον αριθμό σε προσπάθειες.» . Αν ο χρήστης εξαντλήσει και τις 5 προσπάθειες χωρίς να μαντέψει τον αριθμό, θα εμφανίζεται μήνυμα «ΕΧΑΣΕΣ! Ο αριθμός που έψαχνες είναι ο».

ΚΩΔΙΚΑΣ:

```

lesson.py - G:\lesson.py (3.7.0)
File Edit Format Run Options Window Help
import random
x=random.randrange(1,101)
print("ΜΑΝΤΕΨΕ ΤΟΝ ΑΡΙΘΜΟ")
print("Ο αριθμός που πρέπει να μαντέψεις είναι από 1 έως και 100")
print("Έχεις 5 προσπάθειες")
p=1
y=int(input("Μάντεψε τον αριθμό: "))
while y!=x and p<6 :
    p=p+1
    if y<x and p<6:
        print("Λάθος!Ο αριθμός που ψάχνεις είναι μεγαλύτερος")
    if y>x and p<6:
        print("Λάθος!Ο αριθμός που ψάχνεις είναι μικρότερος")
    if p<6 :
        y=int(input("Μάντεψε τον αριθμό: "))
if p<6:
    print("Μπράβο! Βρήκες τον αριθμό σε " , p , " προσπάθειες ")
else :
    print("ΕΧΑΣΕΣ!! Ο αριθμός που έψαχνες είναι ο : " , x)

```

ΕΚΤΕΛΕΣΗ:

```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: G:\lesson.py =====
ΜΑΝΤΕΨΕ ΤΟΝ ΑΡΙΘΜΟ
Ο αριθμός που πρέπει να μαντέψεις είναι από 1 έως και 100
Έχεις 5 προσπάθειες
Μάντεψε τον αριθμό: 50
Λάθος!Ο αριθμός που ψάχνεις είναι μεγαλύτερος
Μάντεψε τον αριθμό: 80
Λάθος!Ο αριθμός που ψάχνεις είναι μικρότερος
Μάντεψε τον αριθμό: 60
Λάθος!Ο αριθμός που ψάχνεις είναι μικρότερος
Μάντεψε τον αριθμό: 55
Λάθος!Ο αριθμός που ψάχνεις είναι μεγαλύτερος
Μάντεψε τον αριθμό: 57
ΕΧΑΣΕΣ!! Ο αριθμός που έψαχνες είναι ο : 56
>>> |

```

ΚΕΦ 6. ΛΙΣΤΕΣ

Η Python χρησιμοποιεί τις λίστες, για να ομαδοποιήσει δεδομένα. Οι λίστες είναι στην ουσία οι πίνακες όπως τους συναντάμε στις άλλες γλώσσες προγραμματισμού.

Έστω μια λίστα $k=[1,2,3,4,5]$. Στη θέση 0 της λίστας υπάρχει ο αριθμός 1, στη θέση 1 της λίστας ο αριθμός 2.....και στη θέση της λίστας 4 ο αριθμός 5.

Παράδειγμα 1:

Λίστα A

ΠΕΤΡΟΣ	ΚΩΣΤΑΣ	ΝΙΚΟΣ	ΜΑΡΙΑ	ΕΛΕΝΗ	ΝΙΚΗ	ΠΑΥΛΟΣ	ΕΛΣΑ	ΠΑΡΗΣ	ΡΕΝΑ
Θέση 0	Θέση 1	Θέση 2	Θέση 3	Θέση 4	Θέση 5	Θέση 6	Θέση 7	Θέση 8	Θέση 9

Η λίστα A έχει 10 θέσεις.

Στον παρακάτω κώδικα δηλώνουμε τη λίστα A. Εμφανίζονται τα δεδομένα στις θέσεις 2 και 8. Μετά τροποποιούνται τα δεδομένα στις θέσεις 4 και 9. Στο τέλος εμφανίζονται όλα τα δεδομένα της λίστας.

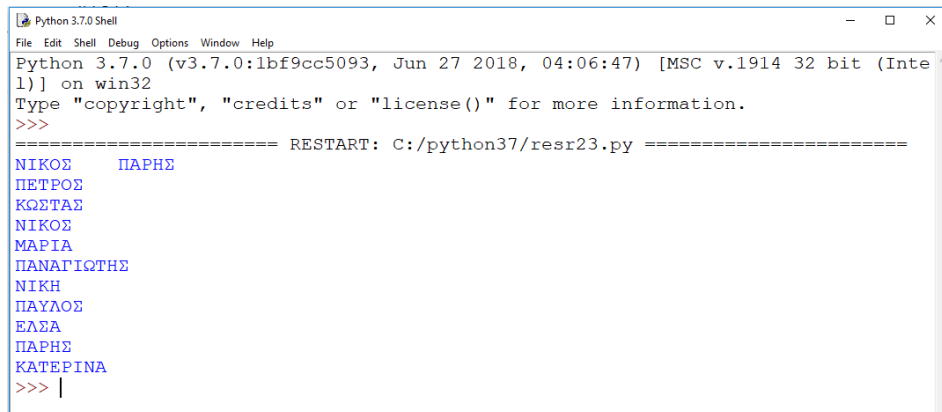
ΚΩΔΙΚΑΣ:

```

resr23.py - C:/python37/resr23.py (3.7.0)
File Edit Format Run Options Window Help
A=["ΠΕΤΡΟΣ", "ΚΩΣΤΑΣ", "ΝΙΚΟΣ", "ΜΑΡΙΑ", "ΕΛΕΝΗ", "ΝΙΚΗ", "ΠΑΥΛΟΣ", "ΕΛΣΑ", "ΠΑΡΗΣ", "ΡΕΝΑ"]
print(A[2], " ", A[8])
A[4]="ΠΑΝΑΓΙΩΤΗΣ"
A[9]="ΚΑΤΕΡΙΝΑ"
for i in range (0,10):
    print(A[i])

```

ΕΚΤΕΛΕΣΗ:




```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/python37/resr23.py =====
ΝΙΚΟΣ ΠΑΡΗΣ
ΠΕΤΡΟΣ
ΚΩΣΤΑΣ
ΝΙΚΟΣ
ΜΑΡΙΑ
ΠΑΝΑΓΙΩΤΗΣ
ΝΙΚΗ
ΠΑΥΛΟΣ
ΕΛΣΑ
ΠΑΡΗΣ
ΚΑΤΕΡΙΝΑ
>>> |

```

Παράδειγμα 2:

ΚΩΔΙΚΑΣ:

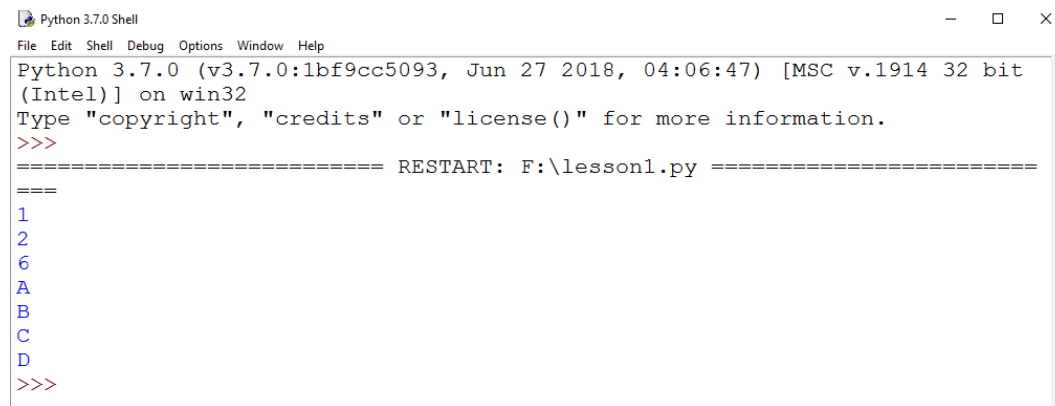


```

lesson1.py - F:\lesson1.py (3.7.0)
File Edit Format Run Options Window Help
k=[1,2,3,4,6]
m=["A", "B", "C", "D"]
print(k[0])
print(k[1])
print(k[4])
for i in range (0,4):
    print(m[i])

```

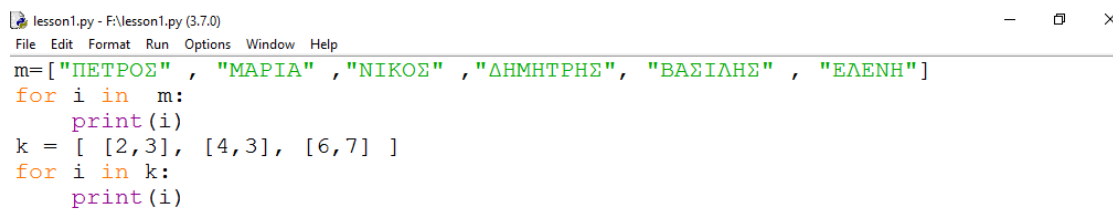
ΕΚΤΕΛΕΣΗ :



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit
(Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: F:\lesson1.py =====
====
1
2
6
A
B
C
D
>>>
```

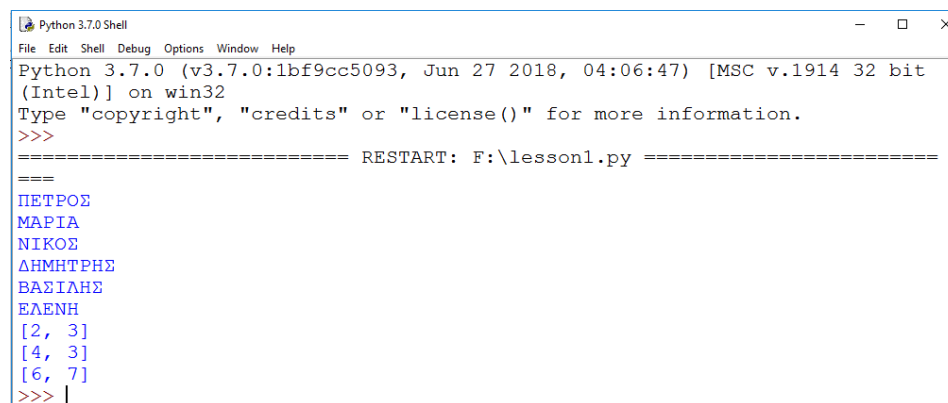
Παράδειγμα 3:

ΚΩΔΙΚΑΣ:



```
lesson1.py - F:\lesson1.py (3.7.0)
File Edit Format Run Options Window Help
m=["ΠΕΤΡΟΣ" , "ΜΑΡΙΑ" ,"ΝΙΚΟΣ" ,"ΔΗΜΗΤΡΗΣ" , "ΒΑΣΙΛΗΣ" , "ΕΛΕΝΗ"]
for i in m:
    print(i)
k = [ [2,3], [4,3], [6,7] ]
for i in k:
    print(i)
```

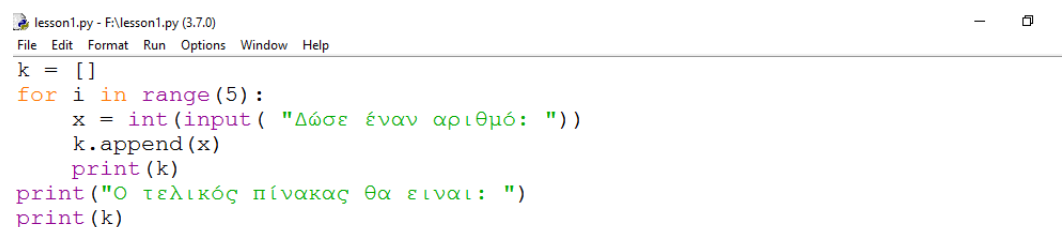
ΕΚΤΕΛΕΣΗ :



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit
(Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: F:\lesson1.py =====
====
ΠΕΤΡΟΣ
ΜΑΡΙΑ
ΝΙΚΟΣ
ΔΗΜΗΤΡΗΣ
ΒΑΣΙΛΗΣ
ΕΛΕΝΗ
[2, 3]
[4, 3]
[6, 7]
>>> |
```

Παράδειγμα 4: Με την εντολή **append** αποθηκεύεται στη λίστα ένα δεδομένο που έχει εισάγει ο χρήστης.

ΚΩΔΙΚΑΣ:



```
lesson1.py - F:\lesson1.py (3.7.0)
File Edit Format Run Options Window Help
k = []
for i in range(5):
    x = int(input( "Δώσε έναν αριθμό: "))
    k.append(x)
    print(k)
print("Ο τελικός πίνακας θα είναι: ")
print(k)
```

ΕΚΤΕΛΕΣΗ:

```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit
(Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: F:\lesson1.py =====
====
Δώσε έναν αριθμό: 12
[12]
Δώσε έναν αριθμό: 1
[12, 1]
Δώσε έναν αριθμό: 4
[12, 1, 4]
Δώσε έναν αριθμό: 15
[12, 1, 4, 15]
Δώσε έναν αριθμό: 6
[12, 1, 4, 15, 6]
Ο τελικός πίνακας θα είναι:
[12, 1, 4, 15, 6]
>>>

```

Παράδειγμα 5:

Να γίνει πρόγραμμα σε ρυθμό όπου χρήστης θα αποθηκεύει 100 ακέραιους αριθμούς σε έναν πίνακα (λίστα). Ο υπολογιστής θα εμφανίζει:

A. Τον μέγιστο αριθμό.

B. Πόσοι αριθμοί του πίνακα είναι μεγαλύτεροι του αριθμού 30.

Γ. Τον μέσο όρο του πίνακα.

ΚΩΔΙΚΑΣ:

```

"lesson1.py - F:\lesson1.py (3.7.0)"
File Edit Format Run Options Window Help
k = []
for i in range(0,100):
    x = int(input( "Δώσε έναν αριθμό: "))
    k.append(x)
max=k[0]
for i in range(0,100):
    if k[i]>max:
        max=k[i]
print("Ο μέγιστος αριθμός του πίνακα είναι ο: " , max)
a=0
for i in range(0,100):
    if k[i]>30:
        a=a+1
print("Στον πίνακα υπάρχουν " , a , "μεγαλύτεροι αριθμοί από το 30")
s=0
for i in range(0,100):
    s=s+k[i]
mo=s/100
print("Ο μέσος όρος του πίνακα είναι: " , mo)

```

Ένα δεδομένο τύπου string μπορεί να φερθεί σαν λίστα

Παράδειγμα 1:

ΚΩΔΙΚΑΣ:

```

"lesson1.py - F:\lesson1.py (3.7.0)"
File Edit Format Run Options Window Help
k = "ΣΧΟΛΕΙΟ"
print(k[0] , k[4])
print("")
k2="ΣΗΜΕΡΑ ΕΧΕΙ ΖΕΣΤΗ"
for i in k2:
    print(i)

```

ΕΚΤΕΛΕΣΗ:

```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit
(Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: F:\lesson1.py =====
====
Σ Ε

Σ
Η
Μ
Ε
Ρ
Α

Ε
Χ
Ε
Ι

Ζ
Ε
Σ
Τ
Η
>>> |

```

Η εντολή len δίνει το μήκος της λίστας ή του string.

Παράδειγμα 2: Θα εμφανιστεί το μήκος του δεδομένου της μεταβλητής k και το μήκος της λίστας k1.

ΚΩΔΙΚΑΣ:

```

lesson1.py - F:\lesson1.py (3.7.0)
File Edit Format Run Options Window Help
k = "ΣΧΟΛΕΙΟ"
k1 = [5, 7, 9, 11, 12]
print(len(k))
print(len(k1))

```

ΕΚΤΕΛΕΣΗ:

```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit
(Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: F:\lesson1.py =====
====
7
5
>>>

```

Παράδειγμα 3: Στο παράδειγμα αυτό βλέπουμε πώς στο παιχνίδι κρεμάλα ο υπολογιστής διαλέγει μία λέξη τυχαία από έναν πίνακα με λέξεις.

ΚΩΔΙΚΑΣ:

```
kremala.py - C:/Users/KOSTAS/Desktop/kremala.py (5.7.0)
File Edit Format Run Options Window Help
import random
words=["ΜΠΑΛΑ", "ΚΑΡΕΚΛΑ", "ΔΩΜΑΤΙΟ", "ΑΥΤΟΚΙΝΗΤΟ", "ΣΧΟΛΕΙΟ"]
lengthwords=len(words)
k=random.randrange(lengthwords)
word=words[k]
print("Η λέξη που επιλέχθηκε είναι η ", word)
print("Η λέξη αποτελείται από ", len(word), " γράμματα")
print("Το πρώτο γράμμα είναι το :", word[0])
|
```

ΕΚΤΕΛΕΣΗ:

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (tags/v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/kremala.py =====
Η λέξη που επιλέχθηκε είναι η ΣΧΟΛΕΙΟ
Η λέξη αποτελείται από 7 γράμματα
Το πρώτο γράμμα είναι το : Σ
>>>
```

ΚΕΦ 7. ΣΥΝΑΡΤΗΣΕΙΣ

Στην ρυθση όπως και στις άλλες γλώσσες προγραμματισμού μπορούν να χρησιμοποιηθούν συναρτήσεις που θα δημιουργήσουν συνθήκες στον προγραμματιστή.

Μια συνάρτηση στην Python έχει τη μορφή:

```
def synartisi():
    Εντολές
```

ή

```
def synartisi(μεταβλητές):
    Εντολές
```

Μετά απλά η συνάρτηση καλείται:

synartisi()

synartisi(4,5)

Πιο αναλυτικά οι συναρτήσεις περιγράφονται από τα παρακάτω παραδείγματα:

Παράδειγμα 1:

ΚΩΔΙΚΑΣ:

```
*kremala.py - C:\Users\KOSTAS\Desktop\kremala.py (3.7.0)*
File Edit Format Run Options Window Help
def team(x):
    print(x, "ολέ ολέ")
team("Liverpool")
team("Man United")
```

ΕΚΤΕΛΕΣΗ :

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/kremala.py =====
Liverpool ολέ ολέ
Man United ολέ ολέ
>>> |
```

Παράδειγμα2: Υπολογισμός εμβαδόν τετραγώνου 3 φορές.

ΚΩΔΙΚΑΣ:

```
*tmeeeee.py - C:\python37\tmeeeee.py (3.7.0)*
File Edit Format Run Options Window Help
def square(a):
    E=a*a
    print("Το εμβαδόν του τετραγώνου είναι: ",E," τετραγωνικά εκατοστά")
for i in range(1,4):
    x=float(input("Δώσε πλευρά τετραγώνου σε εκατοστά "))
    square(x)
```

ΕΚΤΕΛΕΣΗ:

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/kremala.py =====
Δώσε πλευρά τετραγώνου σε εκατοστά 3.3
Το εμβαδόν του τετραγώνου είναι: 10.889999999999999 τετραγωνικά εκατοστά
Δώσε πλευρά τετραγώνου σε εκατοστά 2
Το εμβαδόν του τετραγώνου είναι: 4.0 τετραγωνικά εκατοστά
Δώσε πλευρά τετραγώνου σε εκατοστά 6
Το εμβαδόν του τετραγώνου είναι: 36.0 τετραγωνικά εκατοστά
>>> |
```

Πώς μπορεί να χρησιμοποιήσει μια συνάρτηση μεταβλητές μιας άλλης συνάρτησης:

Με την εντολή `global` μια μεταβλητή γίνεται καθολική.

```
def syna():  
    x=int(input("Give a number "))  
    global k  
    k=x+2  
  
def synb():  
    syna()  
    print(k)  
synb()
```

ΚΕΦ. 8. ΚΛΑΣΕΙΣ ΚΑΙ ΑΝΤΙΚΕΙΜΕΝΑ

Η python είναι μία αντικειμενοστραφής γλώσσα που χρησιμοποιεί κλάσεις και αντικείμενα.

Παράδειγμα 1: Δημιουργία κλάσης `MyClass`.

ΚΩΔΙΚΑΣ:

```
*classes.py - C:/Users/KOSTAS/Desktop/classes.py (3.7.0)*
File Edit Format Run Options Window Help
class MyClass:
    age = 25
    name="kostas"
    flag=True

if MyClass.flag==True :
    print(MyClass.name, " είναι ",MyClass.age , " ετών")

|
```

ΕΚΤΕΛΕΣΗ:

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/classes.py =====
kostas είναι 25 ετών
>>> |
```

Παράδειγμα 2: Δημιουργία αντικειμένου p1.

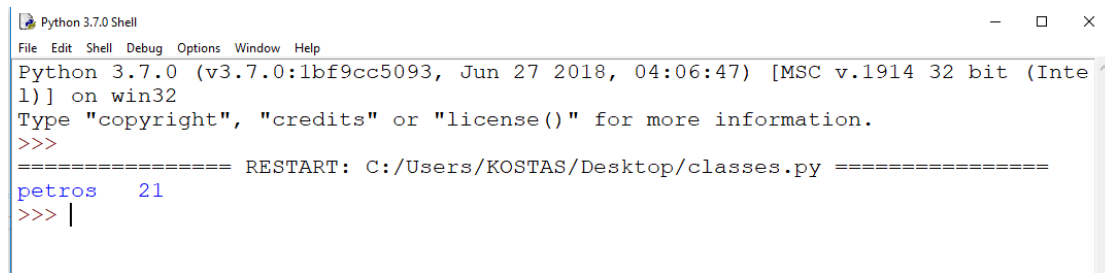
ΚΩΔΙΚΑΣ:

```
*classes.py - C:/Users/KOSTAS/Desktop/classes.py (3.7.0)*
File Edit Format Run Options Window Help
class MyClass:
    age = 25
    name="kostas"
    flag=True

p1=MyClass()
p1.age=21
p1.name="petros"

if p1.flag==True:
    print(p1.name, " ",p1.age)
```

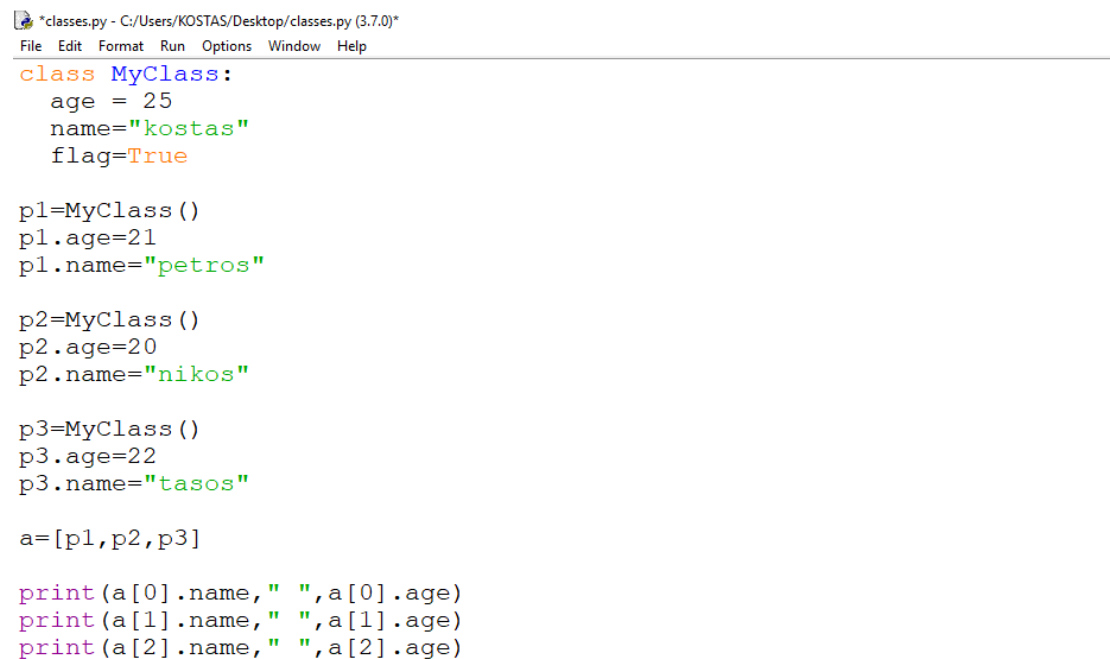
ΕΚΤΕΛΕΣΗ:



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/classes.py =====
petros 21
>>> |
```

Παράδειγμα 3: Εισαγωγή αντικειμένων σε λίστα α.

ΚΩΔΙΚΑΣ:



```
*classes.py - C:/Users/KOSTAS/Desktop/classes.py (3.7.0)*
File Edit Format Run Options Window Help
class MyClass:
    age = 25
    name="kostas"
    flag=True

p1=MyClass()
p1.age=21
p1.name="petros"

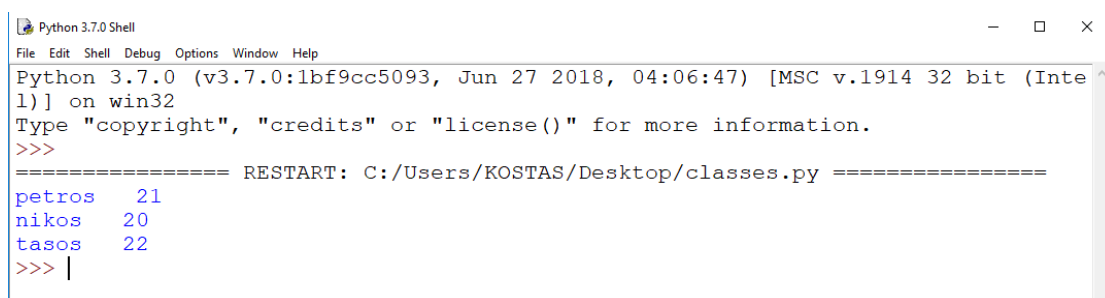
p2=MyClass()
p2.age=20
p2.name="nikos"

p3=MyClass()
p3.age=22
p3.name="tasos"

a=[p1,p2,p3]

print(a[0].name, " ",a[0].age)
print(a[1].name, " ",a[1].age)
print(a[2].name, " ",a[2].age)
```

ΕΚΤΕΛΕΣΗ:



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/classes.py =====
petros 21
nikos 20
tasos 22
>>> |
```

Η συνάρτηση `__init__()`

Χρησιμοποιώντας τη συνάρτηση `__init__()` ακολουθούμε μία άλλη μέθοδο υλοποίησης μιας κλάσης.

Παράδειγμα 4:

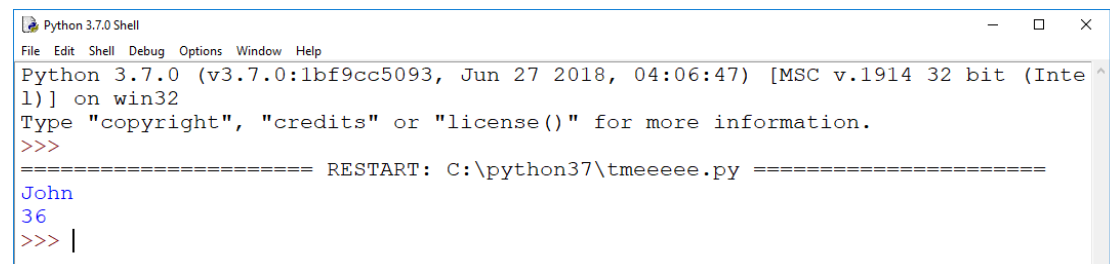
ΚΩΔΙΚΑΣ:

```
*tmeeeeee.py - C:\python37\tmeeeeee.py (3.7.0)*
File Edit Format Run Options Window Help
class Person:
    def __init__(self, name, age):
        self.name = name
        self.age = age

p1 = Person("John", 36)

print(p1.name)
print(p1.age)
```

ΕΚΤΕΛΕΣΗ:

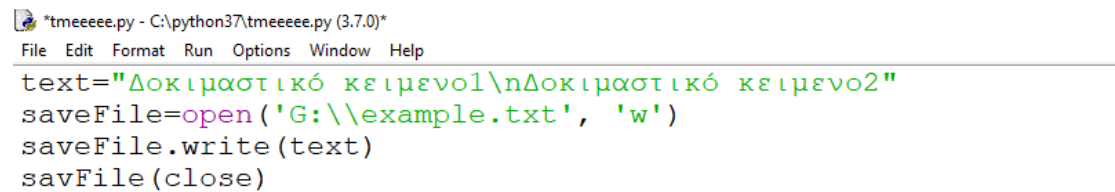


```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\python37\tmeeeeee.py =====
John
36
>>> |
```

ΚΕΦ. 9. ΔΙΑΧΕΙΡΙΣΗ ΑΡΧΕΙΩΝ ΜΕ ΤΗΝ PYTHON.

Παράδειγμα 1: Εγγραφή σε αρχείο.

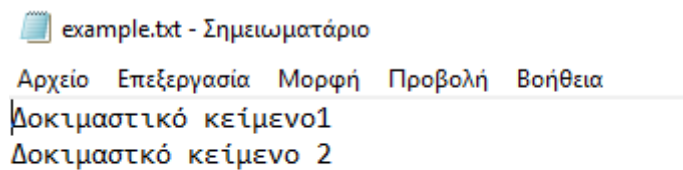
ΚΩΔΙΚΑΣ:



```
*tmeeeeee.py - C:\python37\tmeeeeee.py (3.7.0)*
File Edit Format Run Options Window Help
text="Δοκιμαστικό κείμενο1\nΔοκιμαστικό κείμενο2"
saveFile=open('G:\\example.txt', 'w')
saveFile.write(text)
savFile(close)
```

Μετά την εκτέλεση θα δημιουργηθεί, αν δεν υπάρχει ήδη, το αρχείο στον δίσκο ή στον φάκελο που έχει δηλωθεί.

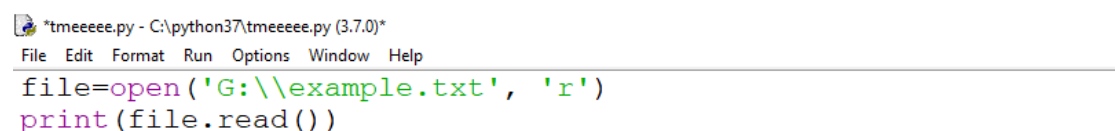
ΕΚΤΕΛΕΣΗ:



```
example.txt - Σημειωματάριο
Αρχείο Επεξεργασία Μορφή Προβολή Βοήθεια
Δοκιμαστικό κείμενο1
Δοκιμαστικό κείμενο 2
```

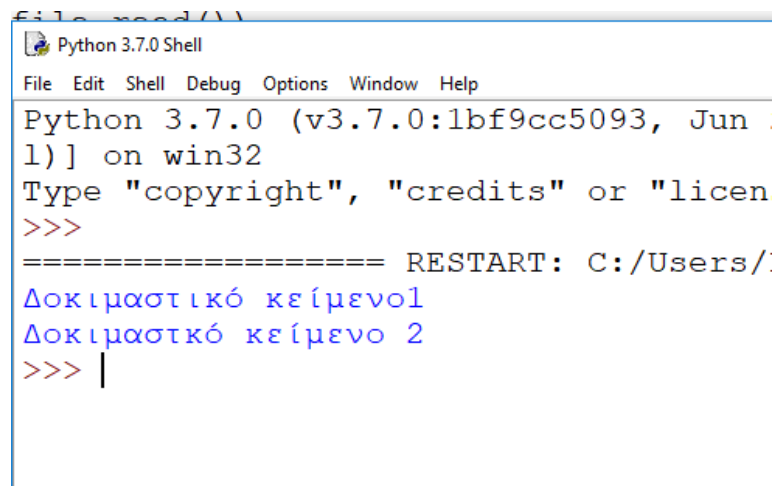
Παράδειγμα 2: Διάβασμα από αρχείο.

ΚΩΔΙΚΑΣ:



```
*tmeeeeee.py - C:\python37\tmeeeeee.py (3.7.0)*
File Edit Format Run Options Window Help
file=open('G:\\example.txt', 'r')
print(file.read())
```

ΕΚΤΕΛΕΣΗ:



```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 1) on win32
Type "copyright", "credits" or "license()" for more
>>>
===== RESTART: C:/Users/...
Δοκιμαστικό κείμενο1
Δοκιμαστικό κείμενο 2
>>> |

```

Αν γράψουμε `print(file.read(3))`, θα εμφανιστούν οι τρεις πρώτοι χαρακτήρες του αρχείου δηλαδή, Δοκ.

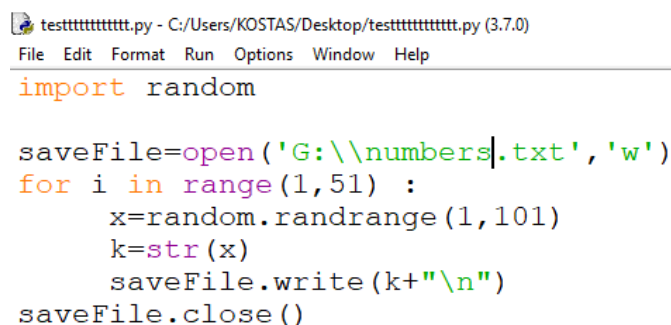
Αν γράψουμε `print(file.readline(2))`, θα εμφανιστεί η δεύτερη σειρά του αρχείου.

Παράδειγμα 3: Δημιουργούμε ένα αρχείο με 50 τυχαίους αριθμούς με όνομα `numbers.txt`. Μετά θα διαβάσουμε τους αριθμούς από το αρχείο και θα εμφανίσουμε το άθροισμα των αριθμών.

ΚΩΔΙΚΑΣ

:

- 1) Βάζουμε στο αρχείο `numbers` 50 τυχαίους αριθμούς τον ένα πάνω από τον άλλον:



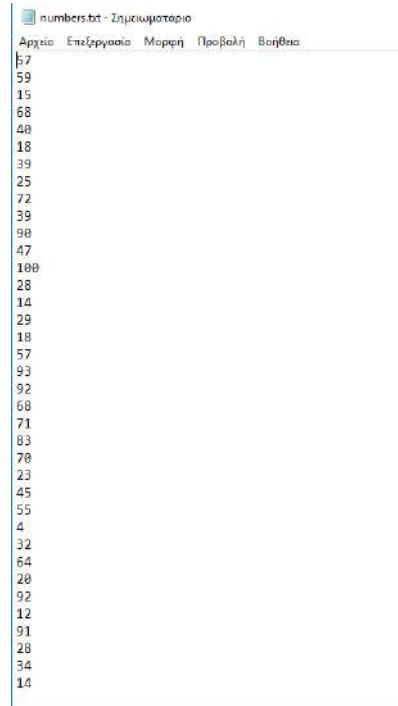
```

testtttttttt.py - C:/Users/KOSTAS/Desktop/testtttttttt.py (3.7.0)
File Edit Format Run Options Window Help
import random

saveFile=open('G:\\numbers.txt','w')
for i in range(1,51) :
    x=random.randrange(1,101)
    k=str(x)
    saveFile.write(k+"\n")
saveFile.close()

```

ΕΚΤΕΛΕΣΗ:



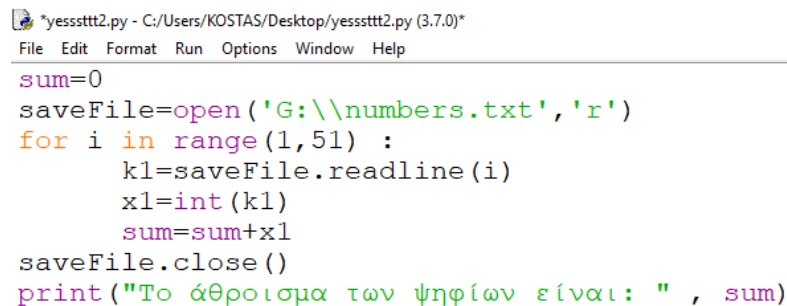
```

numbers.txt - Σημειωματάριο
Αρχείο  Επεξεργασία  Μορφή  Προβολή  Βοήθεια
67
59
15
68
48
18
39
25
72
39
98
47
188
28
14
29
18
57
93
92
68
71
83
78
23
45
55
4
32
64
28
28
92
12
91
28
34
14

```

2) Διαβάζουμε το αρχείο και βρίσκουμε το άθροισμα των αριθμών.

ΚΩΔΙΚΑΣ:

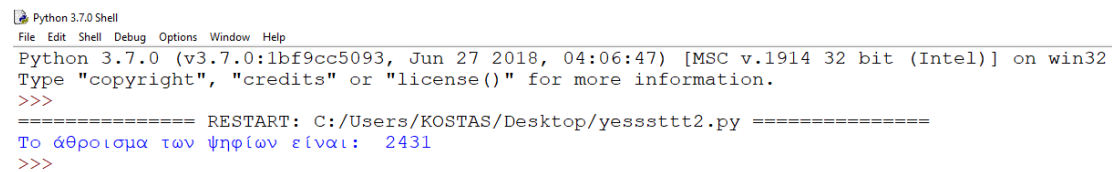


```

*yesssttt2.py - C:/Users/KOSTAS/Desktop/yesssttt2.py (3.7.0)*
File Edit Format Run Options Window Help
sum=0
saveFile=open('G:\\numbers.txt','r')
for i in range(1,51) :
    k1=saveFile.readline(i)
    x1=int(k1)
    sum=sum+x1
saveFile.close()
print("Το άθροισμα των ψηφίων είναι: " , sum)

```

ΕΚΤΕΛΕΣΗ:



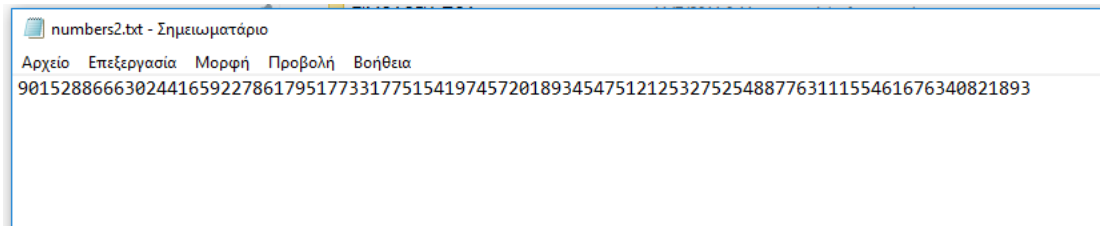
```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/yesssttt2.py =====
Το άθροισμα των ψηφίων είναι: 2431
>>>

```

Αν είχαμε χρησιμοποιήσει κατά την εγγραφή των δεδομένων την εντολή `write` κατά τον ακόλουθο τρόπο: `saveFile.write(k)`, χωρίς δηλαδή το `+"\n"`, τα δεδομένα θα είχαν αποθηκευτεί οριζόντια.

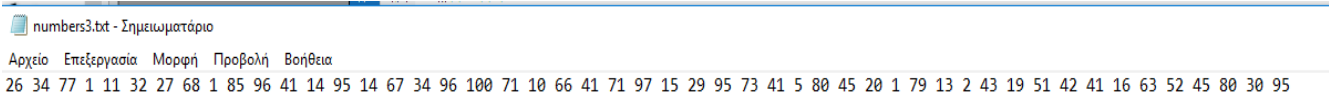
ΕΚΤΕΛΕΣΗ:



```
numbers2.txt - Σημειωματάριο
Αρχείο  Επεξεργασία  Μορφή  Προβολή  Βοήθεια
90152886663024416592278617951773317751541974572018934547512125327525488776311155461676340821893
```

Βλέπουμε ότι δεν υπάρχει κενό μεταξύ των αριθμών. Αν θέλαμε κενό, θα έπρεπε να γράψουμε την εντολή: `saveFile.write(k+" ")`

ΕΚΤΕΛΕΣΗ:



```
numbers3.txt - Σημειωματάριο
Αρχείο  Επεξεργασία  Μορφή  Προβολή  Βοήθεια
26 34 77 1 11 32 27 68 1 85 96 41 14 95 14 67 34 96 100 71 10 66 41 71 97 15 29 95 73 41 5 80 45 20 1 79 13 2 43 19 51 42 41 16 63 52 45 80 30 95
```

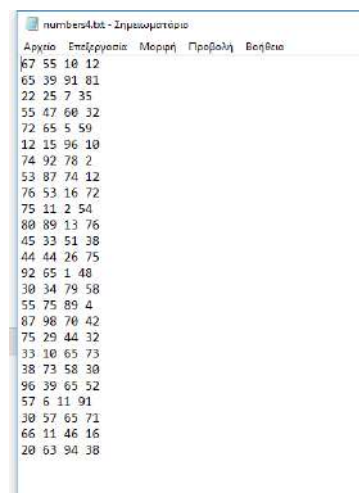
Παράδειγμα 4: Εγγραφή σε αρχείο 100 τυχαίων αριθμών σε 4 στήλες των 25 αριθμών.

```
File Edit Format Run Options Window Help
import random

saveFile=open('G:\\numbers4.txt','w')
for i in range(1,101) :
    x=random.randrange(1,101)
    k=str(x)
    if i%4!=0 :
        saveFile.write(k+" ")
    elif i%4==0 :
        saveFile.write(k+"\n")

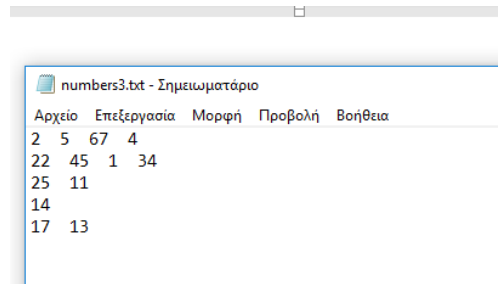
saveFile.close()
```

ΕΚΤΕΛΕΣΗ:



```
numbers4.txt - Σημειωματάριο
Αρχείο  Επεξεργασία  Μορφή  Προβολή  Βοήθεια
67 55 10 12
65 39 91 81
22 25 7 35
55 47 60 32
72 65 5 59
12 15 96 10
74 92 78 2
53 87 74 12
76 53 16 72
75 11 2 54
80 89 13 76
45 33 51 38
44 44 26 75
92 65 1 48
30 34 79 58
55 75 89 4
87 98 70 42
75 29 44 32
33 10 65 73
38 73 58 30
96 39 65 52
57 6 11 91
30 57 65 71
66 11 46 16
20 63 94 38
```

Παράδειγμα 5: Διάβασμα των αριθμών από ένα αρχείο ανά γραμμή και εμφάνιση του μέσου όρου των αριθμών.



ΚΩΔΙΚΑΣ:

```
*symmm.py - C:/Users/KOSTAS/Desktop/symmm.py (3.7.0)*
File Edit Format Run Options Window Help
s=0
κ=0
filehandle=open('F:\\examp\\numbers3.txt', 'r')
for line in filehandle:
    for word in line.split():
        x=int(word)
        s=s+x
        κ=κ+1

print(s/κ)
```

ΕΚΤΕΛΕΣΗ:

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/symmm.py =====
20.0
>>> |
```

Παράδειγμα 6: Διάβασμα από ένα αρχείο λέξεων ανά γραμμή και αποθήκευση σε πίνακα.

ΚΩΔΙΚΑΣ:

```

LIST.py - C:\Users\KOSTAS\Desktop\LIST.py (3.7.0)
File Edit Format Run Options Window Help

a=[]
filehandle=open('F:\\examp\\teams.txt', 'r')
for line in filehandle:
    for word in line.split():
        a.append(word)

for k in a:
    print(k)

```

ΕΚΤΕΛΕΣΗ:

```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\KOSTAS\Desktop\LIST.py =====
ΑΡΗΣ
ΑΕΚ
ΠΑΟ
ΟΣΦΠ
ΠΑΟΚ
ΑΤΡΟΜΗΤΟΣ
ΟΦΗ
ΑΕΛ
ΗΡΑΚΛΗΣ
ΠΑΝΙΩΝΙΟΣ
>>>

```

Παράδειγμα 7: Διάβασμα αριθμών από ένα αρχείο ανά γραμμή και αποθήκευση σε πίνακα. Μετά επεξεργασία του πίνακα, για να υπολογιστεί και να εμφανιστεί ο αριθμός των αρνητικών αριθμών.

```

numbers3.txt - Σημειωματάριο
Αρχείο Επεξεργασία Μορφή Προβολή Βοήθεια
2 -5 67 4
22 -45 1 34
25 11
-14
17 13
-1 -12
37 -45

```

ΚΩΔΙΚΑΣ:

```

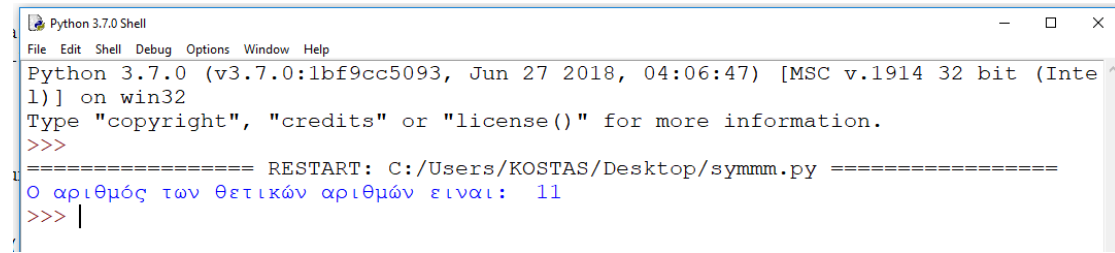
*symmm.py - C:/Users/KOSTAS/Desktop/symmm.py (3.7.0)*
File Edit Format Run Options Window Help

a=[]
k=0
filehandle=open('F:\\examp\\numbers3.txt', 'r')
for line in filehandle:
    for word in line.split():
        x=int(word)
        a.append(x)

for num in a :
    if num>0 :
        k=k+1
print("Ο αριθμός των θετικών αριθμών είναι: ", k)
|

```

ΕΚΤΕΛΕΣΗ:



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/KOSTAS/Desktop/symmm.py =====
Ο αριθμός των θετικών αριθμών είναι: 11
>>> |
```