

ΕΝΤΟΛΗ ΕΠΑΝΑΛΗΨΗΣ FOR

Εντολή Επανάληψης for

2

Άσκηση

Να γραφεί πρόγραμμα το οποίο εμφανίζει στην οθόνη το μήνυμα “Hello World!” 5 φορές.

Πρόγραμμα

```
print “Hello World!”  
print “Hello World!”  
print “Hello World!”  
print “Hello World!”  
print “Hello World!”
```

Εντολή Επανάληψης for

3

Στο πιο κάτω πρόγραμμα παρατηρώ ότι η εντολή **print “Hello World!”** επαναλαμβάνεται **5 φορές**.

Πρόγραμμα

```
1  print “Hello World!”  
2  print “Hello World!”  
3  print “Hello World!”  
4  print “Hello World!”  
5  print “Hello World!”
```

Εντολή Επανάληψης for

4

Πρόγραμμα

```
1  print "Hello World!"
2  print "Hello World!"
3  print "Hello World!"
4  print "Hello World!"
5  print "Hello World!"
6  print "Hello World!"
7  print "Hello World!"
8  print "Hello World!"
9  print "Hello World!"
10 print "Hello World!"
11 print "Hello World!"
12 print "Hello World!"
13 print "Hello World!"
14 print "Hello World!"
15 print "Hello World!"
16 print "Hello World!"
17 print "Hello World!"
18 print "Hello World!"
19 print "Hello World!"
20 print "Hello World!"
```

Αν θέλω το μήνυμα "Hello World!" να εμφανίζεται στην οθόνη του χρήστη 20 φορές θα πρέπει μέσα στο πρόγραμμα να γράψω την ίδια εντολή, **print "Hello World!"**, **20 φορές** κάτι που είναι ασύμφορο.

Εντολή Επανάληψης for

5

Εδώ έρχεται να βοηθήσει η εντολή for μέσω της οποίας λέμε στον υπολογιστή πόσες φορές θέλουμε να επαναλάβει την εντολή που θα του δώσουμε.

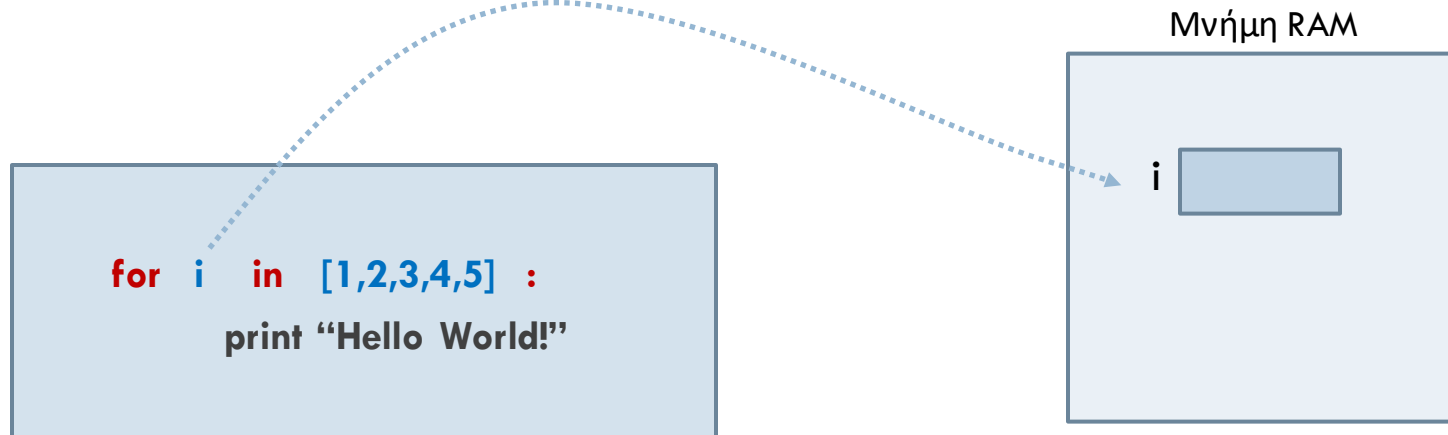
Επιστρέφοντας στο αρχικό μας παράδειγμά αντί να γράψουμε μέσα στο πρόγραμμα 5 φορές την εντολή `print "Hello World!"` μπορούμε να χρησιμοποιήσουμε την πιο κάτω έκφραση που είναι ισοδύναμη:

```
for i in [1,2,3,4,5] :  
    print "Hello World!"
```

Εντολή Επανάληψης for

6

Στην πιο κάτω εντολή το **i** είναι μια **μεταβλητή** (μια θέση μέσα στη μνήμη την οποία δεσμεύει ο υπολογιστής και την ονομάζει i). Δεξιά από τη λέξη in βρίσκεται μια **λίστα τιμών**.



Εντολή Επανάληψης for

7

Η μεταβλητή αυτή θα παίζει το ρόλο του **μετρητή**. Ο υπολογιστής θα ξεκινήσει να βάζει μέσα στη μεταβλητή μία – μία διαδοχικά τις τιμές της δεξιάς λίστας. Κάθε φορά που η μεταβλητή θα έχει μία τιμή ο υπολογιστής θα μπαίνει στην εσοχή και θα εκτελεί την εντολή. Το πλήθος των τιμών της λίστας καθορίζει και το πλήθος των επαναλήψεων (δηλ. πόσες φορές θα επαναληφθεί η εντολή που είναι στην εσοχή).



Εντολή Επανάληψης for

8

Η μεταβλητή αυτή θα παίζει το ρόλο του **μετρητή**. Ο υπολογιστής θα ξεκινήσει να βάζει μέσα στη μεταβλητή μία – μία διαδοχικά τις τιμές της δεξιάς λίστας. Κάθε φορά που η μεταβλητή θα έχει μία τιμή ο υπολογιστής θα μπαίνει στην εσοχή και θα εκτελεί την εντολή. Το πλήθος των τιμών της λίστας καθορίζει και το πλήθος των επαναλήψεων (δηλ. πόσες φορές θα επαναληφθεί η εντολή που είναι στην εσοχή).



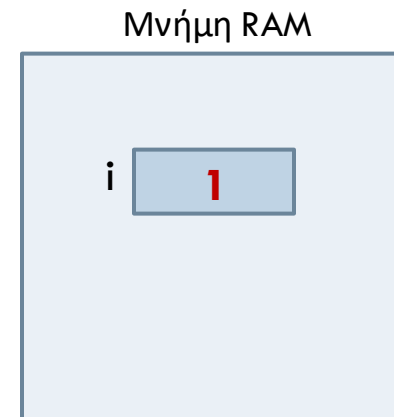
Εντολή Επανάληψης for

9

Έτσι την πρώτη φορά η μεταβλητή **i** θα πάρει την τιμή **1** που είναι και η πρώτη τιμή της λίστας. Ο υπολογιστής μπαίνει στην εσοχή και εκτελεί για πρώτη φορά την εντολή `print "Hello World!"` (1^η επανάληψη) .



```
for i in [1,2,3,4,5] :  
    print "Hello World!"
```



Εντολή Επανάληψης for

10

Στην οθόνη του χρήστη εμφανίζεται για πρώτη φορά το μήνυμα Hello World!



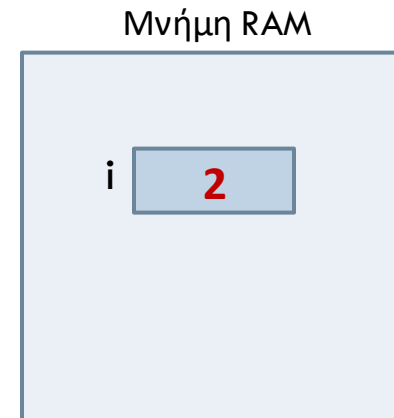
Εντολή Επανάληψης for

11

Στη συνέχεια ο υπολογιστής παίρνει από τη λίστα τη δεύτερη τιμή το **2** και την εκχωρεί στη μεταβλητή. Στη συνέχεια μπαίνει στην εσοχή και εκτελεί για δεύτερη φορά την εντολή `print "Hello World!"` (2^η επανάληψη).



```
for i in [1,2,3,4,5] :  
    print "Hello World!"
```



Εντολή Επανάληψης for

12

Στην οθόνη του χρήστη εμφανίζεται για δεύτερη φορά το μήνυμα Hello World!



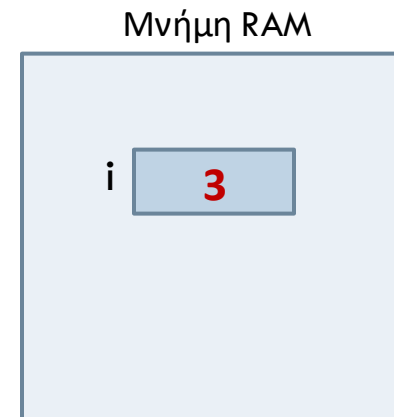
Εντολή Επανάληψης for

13

Στη συνέχεια ο υπολογιστής παίρνει από τη λίστα τη τρίτη τιμή το **3** και την εκχωρεί στη μεταβλητή. Στη συνέχεια μπαίνει στην εσοχή και εκτελεί για τρίτη φορά την εντολή `print "Hello World!"` (3^η επανάληψη).



```
for i in [1,2,3,4,5] :  
    print "Hello World!"
```



Εντολή Επανάληψης for

14

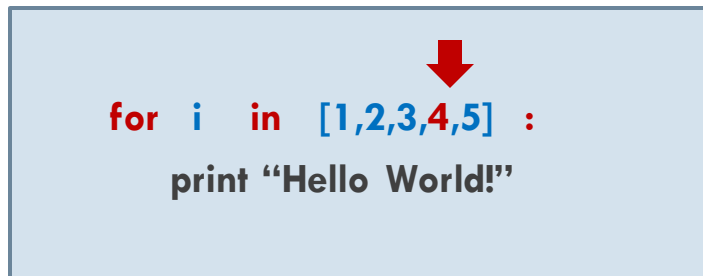
Στην οθόνη του χρήστη εμφανίζεται για τρίτη φορά το μήνυμα Hello World!



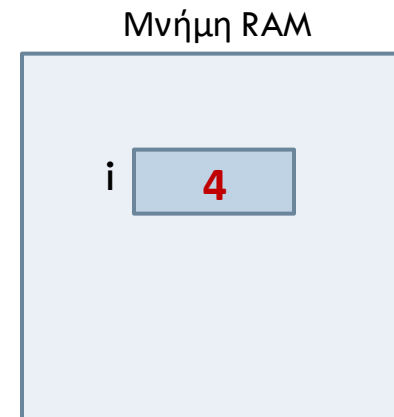
Εντολή Επανάληψης for

15

Στη συνέχεια ο υπολογιστής παίρνει από τη λίστα τη τέταρτη τιμή το **4** και την εκχωρεί στη μεταβλητή. Στη συνέχεια μπαίνει στην εσοχή και εκτελεί για τέταρτη φορά την εντολή print “Hello World!” (4^η επανάληψη).



```
for i in [1,2,3,4,5] :  
    print “Hello World!”
```



Εντολή Επανάληψης for

16

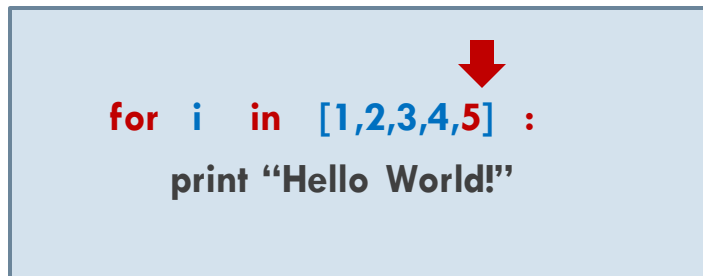
Στην οθόνη του χρήστη εμφανίζεται για τέταρτη φορά το μήνυμα Hello World!



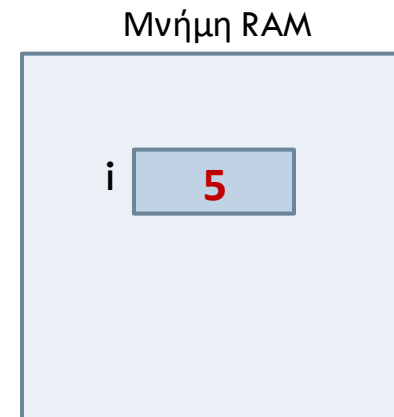
Εντολή Επανάληψης for

17

Τέλος ο υπολογιστής παίρνει από τη λίστα την πέμπτη τιμή το **5** και την εκχωρεί στη μεταβλητή. Στη συνέχεια μπαίνει στην εσοχή και εκτελεί για πέμπτη φορά την εντολή `print "Hello World!"` (5^η επανάληψη).



```
for i in [1,2,3,4,5] :  
    print "Hello World!"
```



Εντολή Επανάληψης for

18

Στην οθόνη του χρήστη εμφανίζεται για πέμπτη φορά το μήνυμα Hello World! και η εντολή **τερματίζει.**



Εντολή Επανάληψης for

19

Προσέξτε ότι το **πλήθος των αριθμών της λίστας** καθορίζει και το **πλήθος των επαναλήψεων**. Στο συγκεκριμένο παράδειγμα οι επόμενες εντολές έχουν όλες το ίδιο αποτέλεσμα.

```
for i in [1,2,3,4,5] :  
    print "Hello World!"
```

```
for i in [0,1,2,3,4] :  
    print "Hello World!"
```

```
for i in [8,15,3,-1,6] :  
    print "Hello World!"
```



Εντολή Επανάληψης for

20

Συνήθως αριθμώ τα στοιχεία της λίστας ξεκινώντας από το 1 και συνεχίζω αυξάνοντας κατά 1. Σταματώ σ' εκείνο το νούμερο που αντιπροσωπεύει το σύνολο των επαναλήψεων που θέλω να γίνουν. Π.χ. για 5 επαναλήψεις χρησιμοποιώ τη λίστα [1,2,3,4,5].

Σ' αυτή τη λίστα ο κάθε αριθμός αντιπροσωπεύει τον αριθμό της επανάληψης (π.χ. 1^η, 2^η, 3^η κλπ επανάληψη).

Κάποια προγράμματα κάνουν χρήση της τρέχουσα τιμής του μετρητή i στις εντολές που βρίσκονται μέσα στην εσοχή (στις εντολές που πρέπει να εκτελούνται σε κάθε επανάληψη). Σε αυτά τα προγράμματα θα δούμε ότι πρέπει να προσαρμόζουμε κατάλληλα τα νούμερα της λίστας.

Εντολή Επανάληψης for

21

Αν τώρα θέλω να εμφανίσω **10** φορές το προηγούμενο μήνυμα θα πρέπει να γράψω την πιο κάτω εντολή βάζοντας το μετρητή **i** να διατρέξει τη λίστα τιμών **[1,2,3,4,5,6,7,8,9,10]**.

```
for i in [1,2,3,4,5,6,7,8,9,10] :  
    print "Hello World!"
```

Εντολή Επανάληψης for

22

Κι αν θέλω να εμφανίσω **20** φορές το προηγούμενο μήνυμα θα πρέπει να γράψω την πιο κάτω εντολή βάζοντας το μετρητή **i** να διατρέξει τη λίστα τιμών **[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]**

```
for i in [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20] :  
    print "Hello World!"
```

Εντολή Επανάληψης for

23

Κι αν θέλω να εμφανίσω **100** φορές το προηγούμενο μήνυμα ?

Συνάρτηση range()

24

Αντί να γράψω τη λίστα κάνω χρήση της έτοιμης συνάρτησης **range()**. Πρόκειται για έναν έτοιμο κώδικα τον οποίο καλώ κάθε φορά που θέλω μέσα στο πρόγραμμά μου να χρησιμοποιήσω μία λίστα με πολλούς αριθμούς.

Η συνάρτηση **καλείται με το όνομά της**. Μέσα στις παρενθέσεις δίνουμε με τη σειρά **3** νούμερα:

1. το νούμερο από το οποίο θέλουμε να **ξεκινά** η λίστα,
2. το νούμερο στο οποίο θέλουμε να **καταλήγει** (**ΠΡΟΣΟΧΗ!** **πρέπει να δίνω πάντα 1 παραπάνω** – π.χ. αν θέλω το καταληκτικό νούμερο να είναι το 20 θα δίνω 21, αν θέλω το 100 θα δίνω 101) και
3. το **βήμα** με το οποίο θα αυξάνουν τα νούμερα της λίστας

Π.χ. αν θέλω τη λίστα **[1,2,3,4,5,6,7,8,9,10]** θα καλέσω τη συνάρτηση ως εξής: **range(1,11,1)**

αν θέλω τη λίστα **[1,2,3,...,100]** θα καλέσω τη συνάρτηση ως εξής: **range(1,101,1)**

Η συνάρτηση δημιουργεί μια λίστα με βάση τα νούμερα που της έχω προσδιορίσει μέσα στην παρένθεση και την επιστρέφει πίσω στο πρόγραμμα στο σημείο που βρισκόταν η κλήση στη συνάρτηση, αντικαθιστώντας την με τη λίστα.

Εντολή Επανάληψης for

25

Στην εντολή for αντί να γράψω τη λίστα τοποθετώ την κλήση στη συνάρτηση range().

```
for i in range(1,21,1) :  
    print "Hello World!"
```

Η συνάρτηση `range(1,21,1)` παράγει τη λίστα `[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]` η οποία στη συνέχεια επιστρέφεται εκεί που βρισκόταν η κλήση και η εντολή παίρνει την τελική μορφή. Το μήνυμα εμφανίζεται 20 φορές στην οθόνη.

```
for i in [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20] :  
    print "Hello World!"
```

Σύνταξη της Εντολής

26

- Όταν ξέρω ότι μια ομάδα εντολών πρέπει να επαναληφθεί N φορές τότε χρησιμοποιώ την εντολή `for`. Δηλαδή αντί να γράψω N φορές τις ίδιες εντολές, τις γράφω μια φορά και τις κλείνω μέσα στο `for loop`.
- Η εντολή **for** κάνει χρήση της συνάρτησης **range()** για τον καθορισμό των επαναλήψεων.

```
for onoma_metavlitis in range(αρχή, μέχρι, βήμα) :  
    Εντολή_1  
    Εντολή_2  
    .....  
    Εντολή_ν
```