

Python: Η εντολή επανάληψης `for i in range ()`

Ας δούμε πως δουλεύει η εντολή της Python **`for i in range ()`** μέσα από παραδείγματα:

- Η **`range ()`** είναι μια συνάρτηση της Python που φτιάχνει μια λίστα με αριθμούς.

Για παράδειγμα:

`range (5)` σημαίνει: "Δώσε μου τους αριθμούς από το 0 έως το 4" (Δηλαδή 5 αριθμούς συνολικά, ξεκινώντας από το 0).

- **`for i in range (5):`**
 - **`for:`** Είναι η λέξη κλειδί που δηλώνει το βρόχο επανάληψης.
 - Η **`range(5)`** δίνει τους αριθμούς: 0, 1, 2, 3, 4.
 - Η μεταβλητή **`i`** ξεκινά από το **0** (δλδ τον πρώτο αριθμό που δίνει η `range`) και σε κάθε επανάληψη, αλλάζει τιμή και παίρνει κάθε αριθμό που δημιουργεί η `range` στη σειρά, μέχρι να φτάσει στο **4**.

Παράδειγμα:

```
for i in range(5)
    print(i)
```

Οπτικοποίηση αποτελέσματος που δίνει ο παραπάνω κώδικας:

Επανάληψη	Τιμή της <code>i</code>	Τι τυπώνει ο κώδικας
1η	0	0
2η	1	1
3η	2	2
4η	3	3
5η	4	4

Εξάσκηση:

1) Να δημιουργήσετε πρόγραμμα σε Python ώστε να εμφανίζεται στην οθόνη του υπολογιστή **12 φορές** η φράση «Είναι ώρα για διάλειμμα!».

2) Παρατηρήστε τον παρακάτω κώδικα. Τι αναμένεται να εμφανιστεί στην οθόνη μετά την εκτέλεση του συγκεκριμένου προγράμματος;

```
1 #Start code here
2 for i in range(6):
3     print(i*2)
4
```

Γράψτε τον συγκεκριμένο κώδικα στον editor της Python και τρέξτε τον ώστε να ελέγξετε την απάντησή σας.

3) Χρησιμοποιείτε κατάλληλα την εντολή επανάληψης **for i in range()** ώστε να εμφανιστούν στην οθόνη οι αριθμοί **από 0 έως και 49**.

4) Μεταφέρετε τον παρακάτω κώδικα στον editor της Python.

- Τι θα εμφανιστεί στην οθόνη του υπολογιστή;
- Ποιος ο ρόλος του αριθμητικού τελεστή ****** ;
- Συμπληρώστε κατάλληλα τον κώδικα ώστε στο τέλος να εμφανίζεται στην οθόνη μια μικρή επεξήγηση σχετικά με τον ρόλο του αριθμητικού τελεστή ****** .

```
i = 4
print(i)
print(i ** 2)
```

5) Μεταφέρετε τον διπλανό κώδικα στον editor της Python.

- Σε τι είδος τιμών δείχνει η μεταβλητή **m**;
- Τι θα εμφανιστεί στην οθόνη; Ποια είναι η αρχική και ποια η τελική τιμή της μεταβλητής **m**;
- Ποια είναι τελικά η λειτουργία της εντολής **m+=1** ;
- Τι προσθήκες/ διορθώσεις πρέπει να κάνετε στον κώδικα ώστε να εμφανίζονται στην οθόνη όλοι αριθμοί **από 1 έως και 100**;

```
1 #Start code here
2 m = 1
3 print(m)
4 m += 1
5 print(m)
6 m += 1
7 print(m)
```

6) Παρατηρήστε τους παρακάτω κώδικες και εξηγήστε τι κάνουν.

A

```
2 print (1)
3 print(2)
4 print (3)
```

B

```
1 #Start code here
2 z = 1
3 v for i in range(3):
4     print(z)
5     z += 1
6
```

Αν σας ζητάγανε να δημιουργήσετε πρόγραμμα που να εμφανίζει τους 50 πρώτους αριθμούς, ξεκινώντας από το 1, θα επιλέγατε, με τις κατάλληλες αλλαγές/ προσθήκες, τον **A** ή τον **B** τρόπο και γιατί;

7) Μεταφέρετε τον παρακάτω κώδικα στον editor της Python.

```
1 #Start code here
2 v for i in range(3):
3     ekfrasi="Πότε θα πάμε εκδρομή;"
4     print (ekfrasi)
```

- Σε τι είδος τιμών δείχνει η μεταβλητή **ekfrasi**;
- Τι θα εμφανιστεί στην οθόνη και πόσες φορές;
- Κάντε στον κώδικα τις απαραίτητες αλλαγές ώστε να εμφανίζεται στην οθόνη το ίδιο αποτέλεσμα, χωρίς τη χρήση της μεταβλητής **ekfrasi**.