



Λίστες

Σε αυτό το σημείο θα μάθουμε τι είναι οι λίστες, πώς ορίζονται, πώς χρησιμοποιούνται και μερικά χρήσιμα tips!

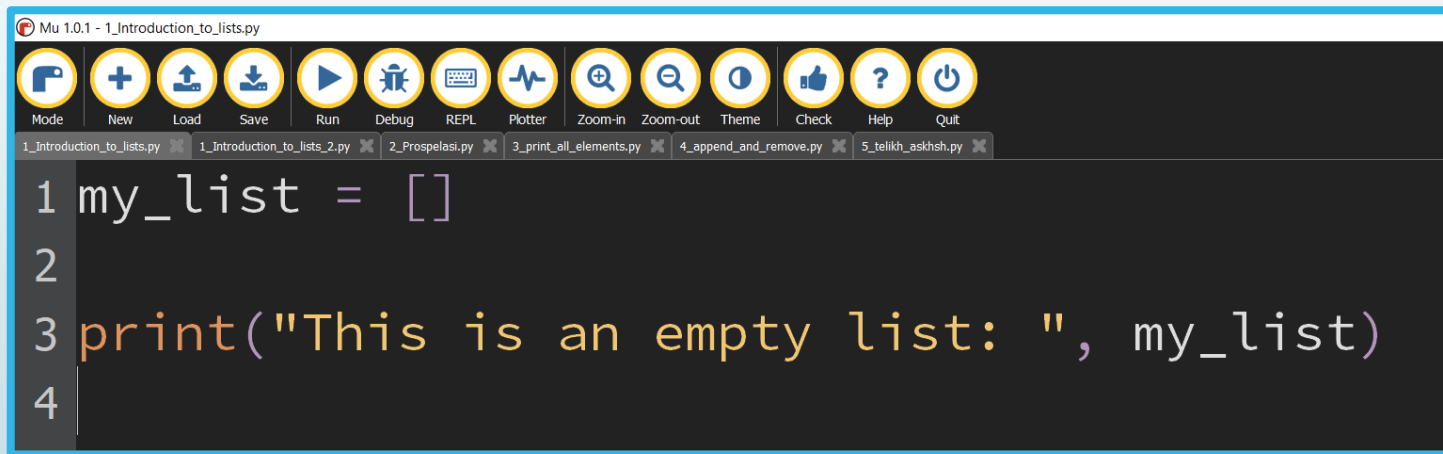


Τι είναι οι λίστες;

Η λίστα είναι μια δομή δεδομένων η οποία περιέχει μια σειρά από δεδομένα.

Σύνταξη - 1

- Ο τρόπος σύνταξης και δημιουργίας μιας νέας, κενής λίστας είναι:



The screenshot shows the Mu Python IDE interface. The title bar reads "Mu 1.0.1 - 1_Introduction_to_lists.py". The toolbar contains icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The file explorer shows several open files: 1_Introduction_to_lists.py, 1_Introduction_to_lists_2.py, 2_Prospeksi.py, 3_print_all_elements.py, 4_append_and_remove.py, and 5_telikh_askhsh.py. The main editor area displays the following Python code:

```
1 my_list = []  
2  
3 print("This is an empty list: ", my_list)  
4
```

Σύνταξη - 2

- Μπορούμε να αρχικοποιήσουμε μια λίστα και ταυτόχρονα να τις προσθέσουμε περιεχόμενο!



The screenshot shows the Mu Python IDE interface. The title bar reads "Mu 1.0.1 - 1_Introduction_to_lists_2.py". The toolbar contains icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The file explorer shows several open files: 1_Introduction_to_lists.py, 1_Introduction_to_lists_2.py, 2_Prospeksi.py, 3_print_all_elements.py, 4_append_and_remove.py, and 5_telikh_askhsh.py. The main editor displays the following Python code:

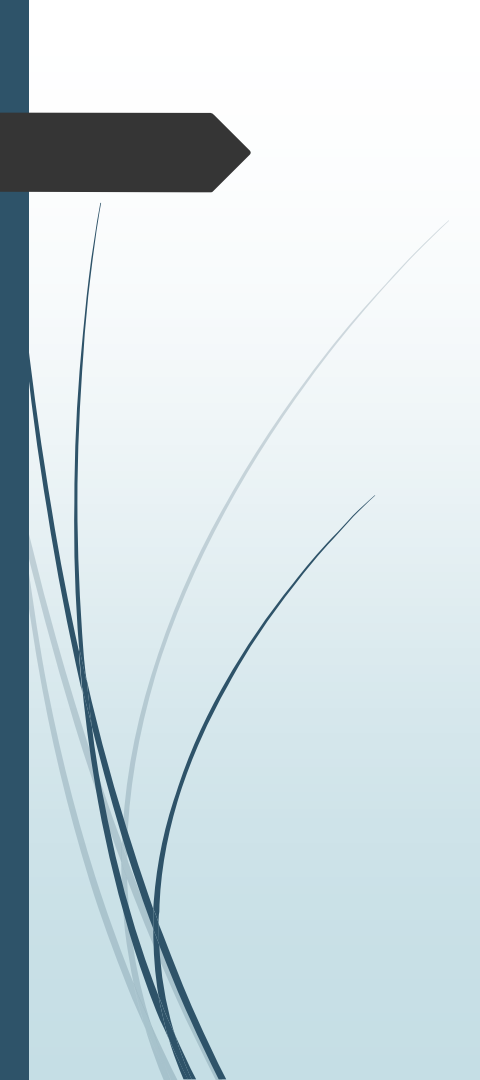
```
1 my_list = [1, 3, 5, 7]
2 print("This is a newly filled list: ", my_list)
3
4 my_list = ["Giorgos" , "Nikos"]
5 print("This is the new list, containing strings", my_list)
6
```



Σημείωση

Οι λίστες μπορούν να περιέχουμε οποιοδήποτε τύπο δεδομένων.

Μπορούμε να εισάγουμε ένα στοιχείο σε μια λίστα ακόμη και αν είναι τύπου `string`, `boolen`, `float` κτλ.



Mu 1.0.1 - 6_Listes_me_diaforous_typous_dedomenon.py

Mode New Load Save Run Debug REPL Plotter Zoom-in Zoom-out Theme Check Help Quit

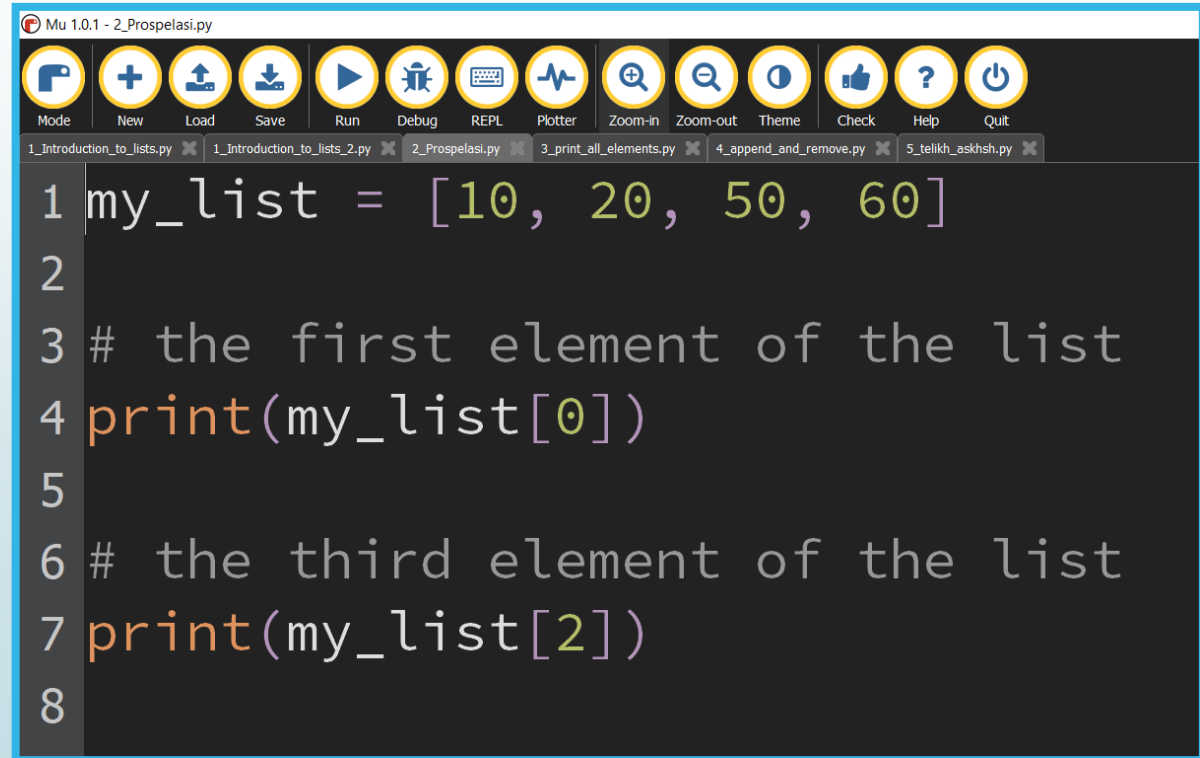
```
1 # Μια λίστα με strings
2
3 shopping_list = ["milk", "eggs", "juice", "soap"]
4
5 # Μια λίστα με booleans
6
7 truth_table = [True, False, True, False, False]
8
9 # Μια λίστα με floats
10
11 samples = [1.2, 0.987, 3.2324, 9.1039]
12
13 # Μια λίστα με διαφορετικούς τύπους δεδομένων
14
15 everything = [1, "String", 0.09, True]
16 for i in everything:
17     print(i)
18
```



Προσπέλαση Στοιχείων - 1

- Αρχικά πρέπει να πούμε πως η αρίθμηση των στοιχείων σε μια λίστα ξεκινά από το **0** και όχι το 1!
- Η σύνταξη που χρησιμοποιούμε για να αναφερθούμε σε ένα συγκεκριμένο στοιχείο στην λίστα είναι η εξής:

Προσπέλαση Στοιχείων - 1



The screenshot shows the Mu Python IDE interface. The title bar reads "Mu 1.0.1 - 2_Prospeksi.py". The toolbar contains icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The file explorer shows several open files: 1_Introduction_to_lists.py, 1_Introduction_to_lists_2.py, 2_Prospeksi.py, 3_print_all_elements.py, 4_append_and_remove.py, and 5_telikh_askhsh.py. The main editor window displays the following Python code:

```
1 my_list = [10, 20, 50, 60]
2
3 # the first element of the list
4 print(my_list[0])
5
6 # the third element of the list
7 print(my_list[2])
8
```


Απεικόνιση Λιστών

<i>0</i>	<i>1</i>	<i>2</i>	<i>3</i>
10	20	50	60

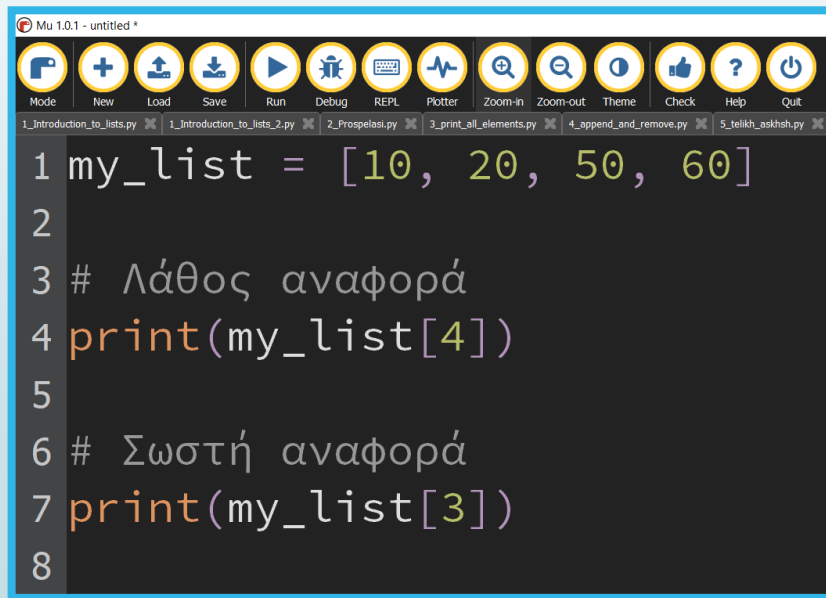


Προσπέλαση Στοιχείων - 2

- **Προσοχή:** Ένα συχνό λάθος που γίνεται είναι ότι όταν θέλουμε να αναφερθούμε στο τελευταίο στοιχείο της λίστας βάζουμε το πλήθος των στοιχείων του πίνακα
- Ακολουθεί παράδειγμα με την λίστα **my_list**

Προσπέλαση Στοιχείων - 2

- Θέλουμε να αναφερθούμε στο τελευταίο στοιχείο της λίστας δηλαδή στον αριθμό 60!

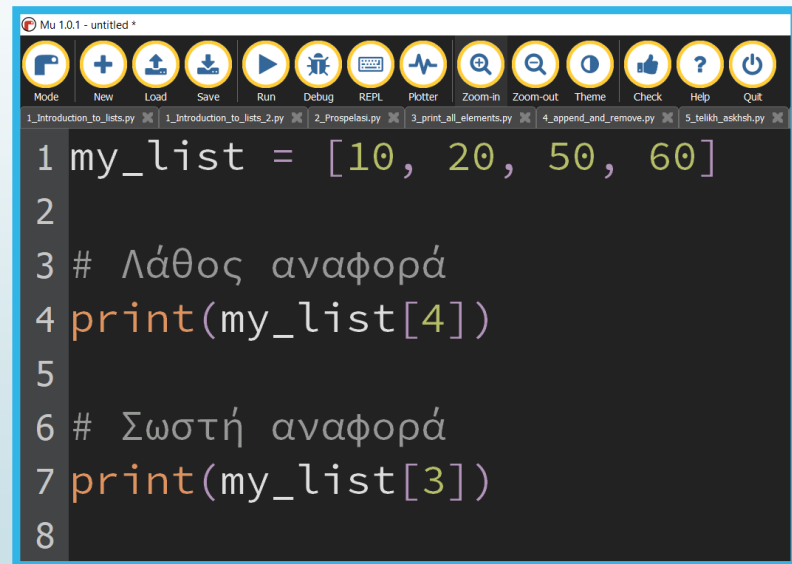


The screenshot shows the Mu Python IDE interface. The title bar reads 'Mu 1.0.1 - untitled *'. The menu bar includes icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The tab bar shows several open files, with '2_Prospeplast.py' selected. The code editor displays the following Python code:

```
1 my_list = [10, 20, 50, 60]
2
3 # Λάθος αναφορά
4 print(my_list[4])
5
6 # Σωστή αναφορά
7 print(my_list[3])
8
```

Προσπέλαση Στοιχείων - 2

- Η πρώτη αναφορά είναι λάθος γιατί ναι μεν θέλουμε να αναφερθούμε στο 4^ο στοιχείο της λίστας, αλλά χρησιμοποιούμε τον αριθμό 3 καθώς η αρίθμηση ξεκινά από το 0.



The screenshot shows the Mu Python IDE interface. The title bar reads "Mu 1.0.1 - untitled *". The menu bar includes: Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The toolbar contains icons for these functions. The file explorer shows several files: 1_Introduction_to_lists.py, 1_Introduction_to_lists_2.py, 2_Prospeksi.py, 3_print_all_elements.py, 4_append_and_remove.py, and 5_telikh_askhsh.py. The main editor window displays the following Python code:

```
1 my_list = [10, 20, 50, 60]
2
3 # Λάθος αναφορά
4 print(my_list[4])
5
6 # Σωστή αναφορά
7 print(my_list[3])
8
```

Προσπέλαση Στοιχείων - 3

- Μπορούμε να διατρέξουμε όλα τα στοιχεία της λίστας χρησιμοποιώντας μια επαναληπτική δομή. Ως επαναληπτική δομή μπορούμε να χρησιμοποιήσουμε την for loop.



The screenshot shows the Mu Python IDE interface. The title bar reads "Mu 1.0.1 - 3_print_all_elements.py *". The toolbar contains icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The file explorer shows several open files, with "3_print_all_elements.py *" selected. The code editor displays the following Python code:

```
1 my_list = [10, 30, 50, 60]
2
3 for i in my_list:
4     print(i)
5
```

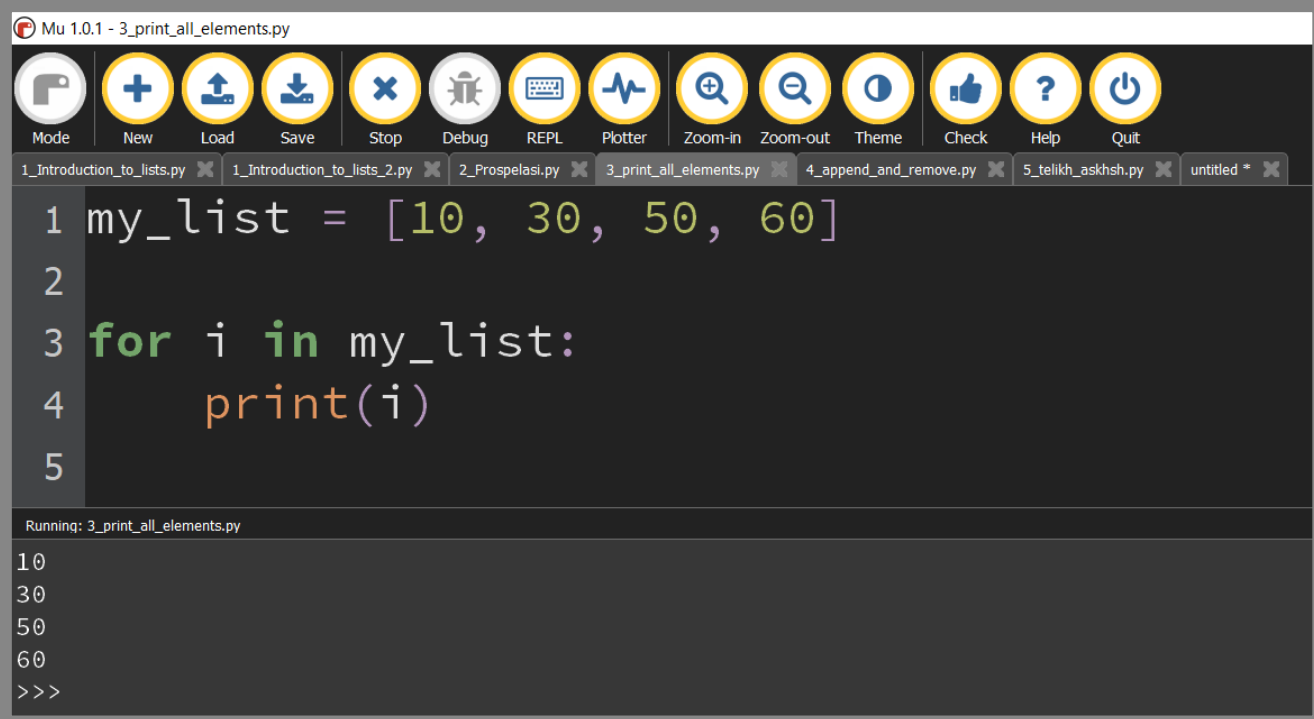


Προσπέλαση Στοιχείων - 3

- **Γιατί συντάσσεται έτσι η for;**
 - Ο λόγος είναι ότι χρειαζόμαστε έναν μετρητή για να μπορούμε να αναφερθούμε κάθε φορά σε ένα μόνο στοιχείο στην λίστα.

Έτσι, ο προηγούμενος αλγόριθμος θα εμφανίσει τα εξής:

Προσπέλαση Στοιχείων - 3



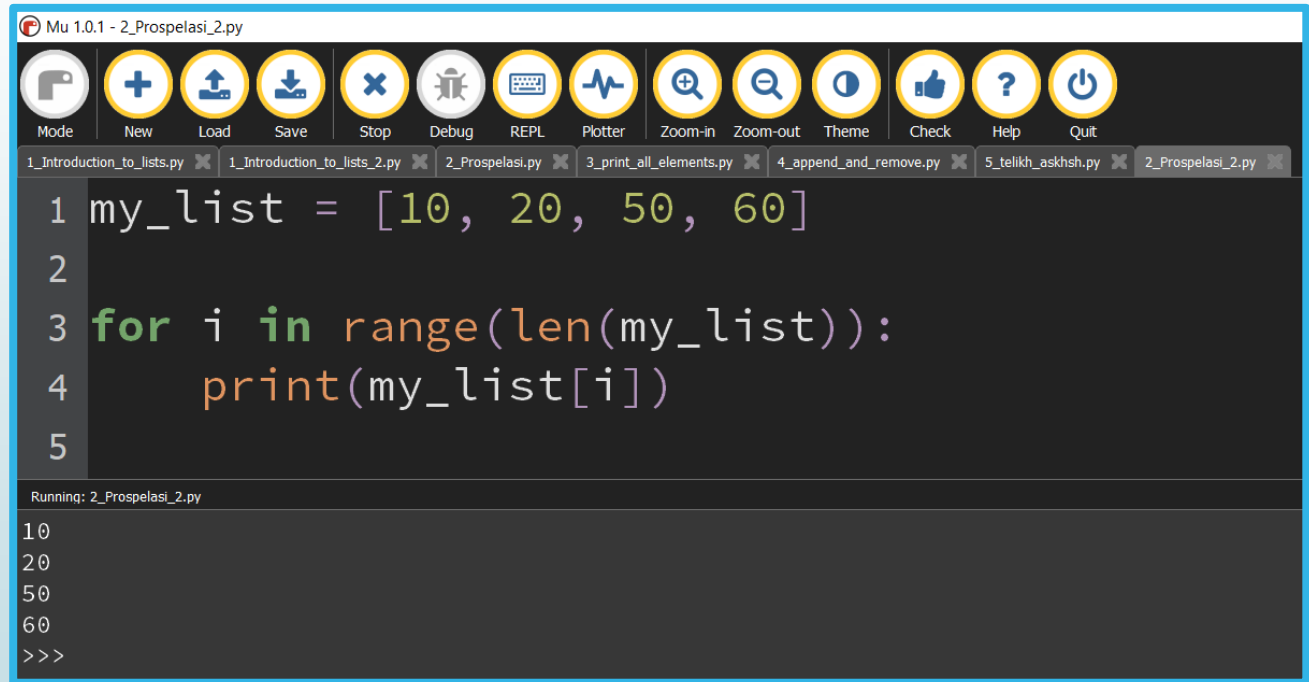
The screenshot displays the Mu Python IDE interface. The title bar reads "Mu 1.0.1 - 3_print_all_elements.py". The top toolbar contains icons for Mode, New, Load, Save, Stop, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The file explorer shows several open files, with "3_print_all_elements.py" selected. The main editor area contains the following Python code:

```
1 my_list = [10, 30, 50, 60]
2
3 for i in my_list:
4     print(i)
5
```

Below the editor, the output console shows the results of running the script:

```
Running: 3_print_all_elements.py
10
30
50
60
>>>
```

Εναλλακτική Προσπέλαση



The screenshot displays the Mu Python IDE interface. The title bar reads "Mu 1.0.1 - 2_Prospektasi_2.py". The top toolbar contains icons for Mode, New, Load, Save, Stop, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. Below the toolbar, a tab bar shows several open files, with "2_Prospektasi_2.py" selected. The main editor area contains the following Python code:

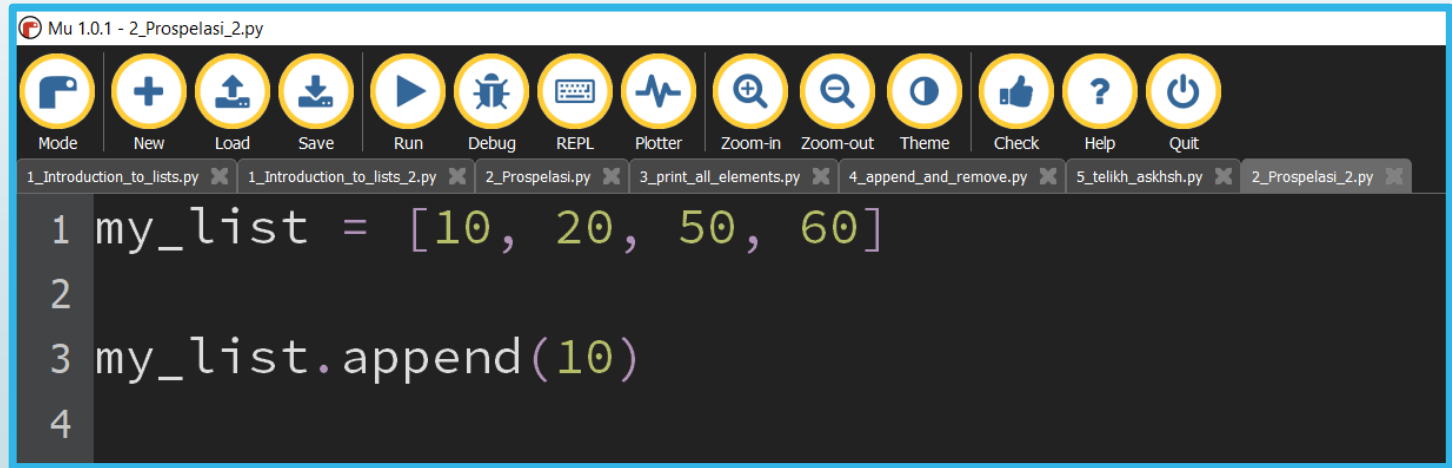
```
1 my_list = [10, 20, 50, 60]
2
3 for i in range(len(my_list)):
4     print(my_list[i])
5
```

Below the editor, a status bar indicates "Running: 2_Prospektasi_2.py". The output console at the bottom shows the execution results:

```
10
20
50
60
>>>
```


Εισαγωγή Στοιχείων - 1

- Θα χρειαστούμε την μέθοδος **append()**.
- Παράδειγμα:



The screenshot shows the Mu Python IDE interface. The title bar reads "Mu 1.0.1 - 2_Prospectasi_2.py". The top toolbar contains icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. Below the toolbar, a tab bar shows several open files, with "2_Prospectasi_2.py" selected. The main editor area displays the following Python code:

```
1 my_list = [10, 20, 50, 60]
2
3 my_list.append(10)
4
```

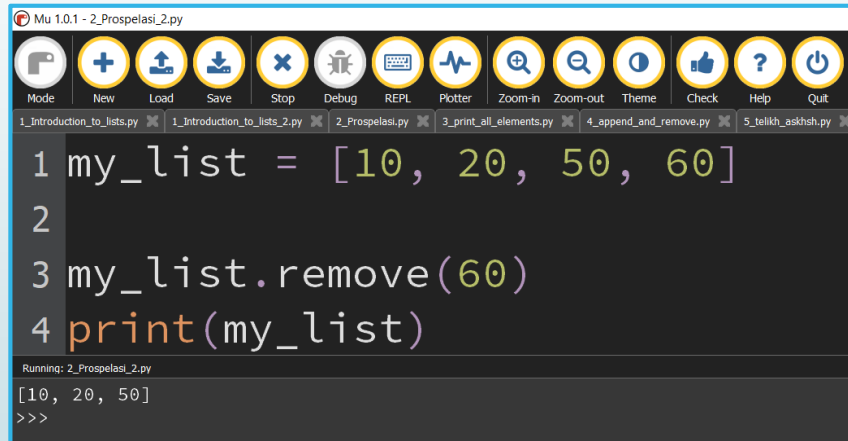
Εισαγωγή Στοιχείων - 2

- Η λίστα μετά την κλήση της συνάρτησης `append()` θα είναι:

```
[10, 20, 50, 60, 10]  
>>>
```

Αφαίρεση Στοιχείων

- Αντίστοιχα υπάρχει η συνάρτηση **remove()**
- Παράδειγμα:



The screenshot shows the MuPython IDE interface. The title bar reads 'Mu 1.0.1 - 2_Prospeasi_2.py'. The menu bar includes Mode, New, Load, Save, Stop, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The toolbar contains icons for these actions. The editor shows the following code:

```
1 my_list = [10, 20, 50, 60]
2
3 my_list.remove(60)
4 print(my_list)
```

Below the editor, the output area shows the result of running the code:

```
Running: 2_Prospeasi_2.py
[10, 20, 50]
>>>
```

- Οπότε το 60 έχει αφαιρεθεί από την λίστα
- **Προσοχή:** Δεν αναφερόμαστε στο στοιχείο που βρίσκεται στην 17^η θέση του πίνακα αλλά στο στοιχείο ίσο με 17