

Μάθημα: Αρχές Προγραμματισμού Υπολογιστών

Κεφάλαιο 5: Δομές Δεδομένων

Φύλλο εργασίας λίστες

Λέξεις-κλειδιά:

Δομή δεδομένων, λίστα, συνάρτηση len(), υπαρξιακός τελεστής in, τελεστής not in, τελεστής : (φέτας, τεμαχισμού), τιμή None, εντολή dir(), εντολή help(), μέθοδοι, append(), insert(), extend(), pop(), remove(), sort(), reverse(), εντολή del, συνάρτηση max(), min(), sum(), αντίγραφο λίστας, κενή λίστα, συνάρτηση list().

1. Ανοίξτε το **Python IDLE**.
2. Στο **shell** πληκτρολογήστε με τη σειρά τις ακόλουθες εντολές και στην αντίστοιχη στήλη συμπληρώστε τα αποτελέσματα που βλέπετε στην οθόνη σας.

Εντολή	Οθόνη	Επεξήγηση
>>>L1=[20,30,40,50,60] >>print L1 >> print len(L1)		Εμφάνιση λίστας <u>Μέγεθος(μήκος) λίστας</u>
>>for item in L1: print item		Εμφάνιση στοιχείων λίστας
>>for i in range(len(L1)): print L1[i]		Εμφάνιση στοιχείων λίστας
>>>30 in L1 >>>35 not in L1		Τελεστής in (ανήκει) Τελεστής not in (δεν ανήκει)
>>>print L1[1:3] >>>print L1[2:] >>>print L1[:2] >>>print L1[:] >>>print L1[::2]		Φέτα(τμήμα) λίστας
>>>L1[0]=10 >>>print L1		Μεταβαλλόμενη (mutable) δομή

<pre>>>>L1[1]=None >>>print L1 >>>L1[1]=20 >>>print L1</pre>		None=Απουσία τιμής
<pre>>>>L2=[] >>>print L2 >>>L2[0]=5 >>>print L2 >>>L2=list() >>>print L2 >>>L2[0]=5 >>> L2=L2+[90,70,80] >>>print L2</pre>		Κενή λίστα Συνένωση λιστών
<pre>>>>L3=3*L2 >>>print L3</pre>		Επανάληψη λίστας
<pre>>>>L4=10*[None] >>>print L4 >>>L4[0]=3 >>>print L4</pre>		Επανάληψη λίστας με None
<pre>>>>dir(list)</pre>		Εμφάνιση όλων των διαθέσιμων μεθόδων μιας λίστας
<pre>>>>help(list)</pre>		Συνοπτική περιγραφή της λειτουργίας μιας μεθόδου λίστας
<pre>>>>L3.append(100) >>>print L3</pre>		<u>Προσθήκη στοιχείου στο τέλος</u>
<pre>>>>L3.insert(5,1) >>>print L3</pre>		<u>Προσθήκη στοιχείου σε συγκεκριμένη θέση</u>

<pre>>>>L3.extend([110,120]) >>>print L3</pre>		<u>Εισαγωγή όλων των στοιχείων μιας λίστας στο τέλος μιας άλλης</u>
<pre>>>>L3.pop() >>>print L3 >>>L3.pop(2) >>>print L3</pre>		<u>Διαγραφή στοιχείου από το τέλος</u> <u>Διαγραφή στοιχείου από συγκεκριμένη θέση</u>
<pre>>>>del L3[3]</pre>		<u>Διαγραφή συγκεκριμένου στοιχείου</u>
<pre>>>>print L3</pre>		
<pre>>>>L3.remove(110) >>>print L3 >>>L3.remove(90) >>>print L3 >>>L3.remove(6) >>>print L3</pre>		<u>Διαγραφή 1ης εμφάνισης στοιχείου</u> Μήνυμα λάθους αν το στοιχείο δεν υπάρχει(ValueError)
<pre>>>>L3.reverse() >>>print L3 >>>L3.sort() >>>print L3 >>>L3.sort(reverse=True) >>>print L3</pre>		<u>Αντιστροφή λίστας</u> <u>Ταξινόμηση λίστας σε αύξουσα σειρά</u> <u>Ταξινόμηση λίστας σε φθίνουσα σειρά</u>
<pre>>>>max(L3) >>>min(L3) >>>sum(L3)</pre>		<u>Μέγιστο στοιχείο λίστας</u> <u>Ελάχιστο στοιχείο λίστας</u> <u>Άθροισμα όλων των στοιχείων λίστας</u>

>>>L2=L3 >>>print L2,L3		<u>Οι μεταβλητές αναφέρονται στο ίδιο αντικείμενο λίστας</u>
>>>id(L2) >>>id(L3)		<u>Διεύθυνση μνήμης αντικειμένου λίστας</u>
>>>L2.append(8) >>>print L2, L3 >>>L3.pop() >>>print L2,L3		
>>>L2=L3[:] >>>id(L2) >>>id(L3) >>>L2.append(9) >>>print L2,L3		<u>Αντίγραφο λίστας</u>
>>>L2=L3+[] >>>id(L2) >>>id(L3) >>>L2.append(12) >>>print L2,L3		Αντίγραφο λίστας
>>>L2.append([7,12]) >>>print L2		Στοιχείο λίστας=Λίστα
>>>print L2[len(L2)-1][1]		Εμφάνιση στοιχείου λίστας που περιέχεται μέσα σε άλλη λίστα

!Συνάρτηση `reversed()`, `sorted()`(αντιστρέφει και ταξινομεί λίστα δημιουργώντας αντίγραφο)

!Μέθοδοι `index()`, `count()` (θέση στοιχείου και πλήθος εμφανίσεων στοιχείου λίστας)