

## Data Type: Lists

What happens if we want to compute the average of 10 or more numbers? It will be silly to ask users to enter 10 or even more values and write many input statements.

The above program can be modified into the following one:

Program	Output
<pre>def sum_numbers():     x = int(input("How many numbers would you like to add?"))     #Allow the user to enter the no of values.     sum = 0     for i in range(0, x):         num = int(input("Enter number " + str(i) + " please:"))         sum = sum + num     avg = sum/x     print("Average:",avg)  sum_numbers()</pre>	<pre>How many numbers would you like to add?5 Enter number 0 please:3 Enter number 1 please:6 Enter number 2 please:3 Enter number 3 please:8 Enter number 4 please:9 Average: 5.8 &gt;&gt;&gt;</pre>

But still, the input values cannot be retained for future recall. To keep the input values, we use the data type *list*.

A *list* refers to a collection of objects; it represents a sequence of data. A list need not be homogeneous; that is, it can hold integer, float and string all on one list.

eg. `>>>firstlist = [2,-5, "a", "b", 9.5]` #the elements of the list appear within [ ] brackets and are separated by commas.

`>>>print(firstlist)`

will show [2, 5, 'a', 'b', 9] as the output

We may access the elements contained in a list via their position within the list.

firstlist	→	2	-5	"a"	"b"	9.5
Position	→	0	1	2	3	4

`firstlist[0] = 2` and `firstlist[3]="b"` but the last element is (n-1), ie. `firstlist[4] = 9.5` instead of `firstlist[5]`.

### List operations and functions:

In interactive mode, enter the following statement and observe its output. Explain what the operator does on list.

Question	Statement	Output	Explanation of the use of the operator
1	<pre>&gt;&gt;&gt;a = [3,7,10] &gt;&gt;&gt; a += 11 &gt;&gt;&gt; print(a)</pre>		What does it do?
2	<pre>&gt;&gt;&gt;a = [3,7,10] &gt;&gt;&gt; a += [11] &gt;&gt;&gt;print(a)</pre>		What does it do?  Explain the difference between Q(1) and Q(2)
3	<pre>&gt;&gt;&gt;a = [3, 7, 10] &gt;&gt;&gt;b = [11, 13, 'spcs'] &gt;&gt;&gt;print(a+b)</pre>		What does it do?  It is the same as ...

4	<pre>&gt;&gt;&gt; b = [3.5, 14] &gt;&gt;&gt; print(b * 5)</pre>		What does it do?
5	<pre>&gt;&gt;&gt; c = ['ABC'] &gt;&gt;&gt; print(c*3)</pre>		What does it do?
6	<pre>&gt;&gt;&gt; a = [2, 6, 9, 1, 13] &gt;&gt;&gt; a[3] = - 8 &gt;&gt;&gt; print(a)</pre>		What does line 2 do?

Operators used in list are \_\_\_\_\_, \_\_\_\_\_ and \_\_\_\_\_.

Functions:

Enter the following program and observe its output. Explain what the function does.

Questions	Program	Output	Function
1	<pre>new_list = [2, 4, 6, 8] num = len(new_list) print(num)</pre>		What does len() do?
2	<pre>#making list a = list(range(0, 10)) print("a = ", a) b = list(range(1, 100, 10)) print("b = ", b) c = list(range(10, -1, -1)) print("c = ", c) d = list(range(-10, 5)) print("d = ", d)</pre>		<p>What does list(range()) do?</p> <p>Format of list(range()) is  list(range(m, n, step)) where  m = _____ (including m)  n = _____ (excluding n) and  step = _____  If there is no st value, by default st is 1</p>

The program on finding the average of 5 numbers can further be modified using list

Program	Output
<pre>def sum_numbers():     num_list = [5, 3, 9, 4, 11]    #assigning 5 values to the list.     sum = 0     for i in num_list:    # read as for element in num_list         sum = sum + i    # same as sum += i         avg = sum/len(num_list)     print("Sum is ", sum)     print("Average is ",avg)  sum_numbers()</pre>	<pre>Sum is 32 Average is 6.4 &gt;&gt;&gt;</pre>

List slicing:

It follows the same as string slicing. The format is

`Sublist=listname[begin: end]`

If alist=[10, 20, 30, 40, 50, 60, 70, 80], then

Statement	Output
<code>print(alist)</code>	
<code>print(alist[0:3])</code>	
<code>print(alist[5:8])</code>	
<code>print(alist[-5:-3])</code>	
<code>print(alist[:4])</code> <i>#if a parameter is missing, by default, it is 0</i>	
<code>print(alist[4:])</code>	
<code>print(alist[-100:3])</code> <i>#if the end value is greater than the length of the list, it is treated as the length of the list</i>	
<code>print(alist[3:100])</code> <i>#if the end value is greater than the length of the list, it is treated as the length of the list</i>	

Program Practice (for submission)

Choose one of the following questions:

1. Write a program using list and functions which counts the even numbers in a list of 5 integers entered by the user. For example, if *list a* contains the elements 8,5,4,-1, and 2, the program would evaluate to 3, since there are three even numbers in the list. The function returns zero if the list is empty.
2. Write a program using list and function which adds up all the positive values in a list of 5 integers entered by the user. For example, if *list a* contains the elements 3,-3,5,2,-1, and 2, the sum would evaluate to 12, since 3+5+2+2 = 12. The function returns zero if the list is empty.