Python Lists

1. Write Python statements to create the following lists:
   1. A list of 5 fruit names and name it as fruit.
   2. A list of 5 city names and name it as City.
   3. A list containing 3 strings and 2 integers and name it as Mixed.
   4. A list containing 3 integers and a list of 2 strings and name it as Box.
   5. A list of 10 random integers each in the range 10 to 50 and name it as R10.
   6. A list of 10 random floating point numbers in the range 10.00 to 49.99 and name it as F10.
   7. A list of 10 values of the expression x2+2x-1, for integer values of x from -4 to 5 and name it as y.
   8. A list of 10 random integers in the range -30 to -20 and name it as C.
   9. A list, named A, containing 3 numbers, and another list, named B, congaing list A and 3 numbers.
   10. A list, named A, containing 3 numbers, and another list, named B, containing list A and another unnamed list of 3 integers.
2. Assuming that A is a list, differentiate between:
   1. A[2] and A[2:]
   2. A[2:] and A[:2]
   3. A[2::] and A[::2]
   4. A\*3 and [A,A,A]
3. Consider list A given below and write the output of the statements that follow:

A=[3,6,7,8,10,12,23,33,16,6,2,1]

|  |  |  |
| --- | --- | --- |
| (i) print(len(A)) |  | (ii) print(A[3]) (iii) print(A[-3]) |
| (iv) print(A[3:]) |  | (v) print(A[-3:]) (vi) print(A[::3]) |
| (vii) print(A[3::]) |  | (viii) print(A[3::-2]) (ix) print(A[-3::-2]) |
| (x) print(A[::-3]) |  | (xi) print(A[3:-3:3]) (xii) print(A[3:-3:-3]) |
| (xiii) print(len(A[3:]))  4. Consider the lists A and B giv follow: |  | (xiv) print(len(A[-3:])) (xv) print(len(A[::-3]))  en below and state the values (True/False) of the expressions that |

A=[5, 3, 8]

B=[A, 8, 66, 45]

|  |  |  |
| --- | --- | --- |
| (i) A in B | (ii) 'A ' in B | (iii) [A] in B (iv) [5, 3, 8] in B |
| (v) 8 in B | (vi) [8] in B | (vii) [A, 8] in B (viii) 66 in B |
| (ix) 45 in B | (x) A not in B |  |

1. Write output of the following code segment:

a=[1,2,3,2,1]

for x in a:

print(a[x],end= '\*')

1. Write output of the following code segment:

A=[x for x in range(2,5)]

B=[-x for x in range(2,5)]

for i in range(len(A)):

print(A[i]+B[i]) A.extend(B)

B.insert(2,3)

for i in B:

if i in A:

A.remove(i) print(A, B) x=A.pop() y=B.pop(0)

print(x, y) print(A,B)

1. Write output of the following code segment: nums=[1,2,7,9,5,7,2,4,5,1,1,2] for i in range(1,10):

if i in nums:

index=nums.index(i) c=nums.count(i)

print(i,'appears in nums',c,'times.',end=' ') print('Its first occurence is at index',index)

1. Consider the lists A and B given below and write the output of the statements that follow:

A=[5, 3, 8]

B=[A, 8, 66, 45, 'A',['A',A], ["nested", "lists"]]

|  |  |  |
| --- | --- | --- |
| (i) print(A) |  | (ii) print(B) (iii) print(A[0]) |
| (iv) print(A[0:]) |  | (v) print(A[:0]) (vi) print(B[0]) |
| (vii) print(B[0][1]) |  | (viii) print(B[0][1:]) (ix) print(B[4]) |
| (x) print(B[5])  . Write output of the following import copy values=[5,3,9,7,1, values1=values.cop values2=copy.copy( values3=copy.deepc values.sort() print(values) print(values1) print(values2) print(values3)   1. Write output of the following import copy a=[7,1,2] A=[5,3,9,a,4] B=A.copy() C=copy.copy(A) D=copy.deepcopy(A) a.sort()   print(a) print(A) print(B) print(C) print(D) B[3][1]=10  D[3][1]=55  print(a) print(A) print(B) print(C) print(D) |  | (xi) for i in B: (xii) for i in A: print(I, end='\*') print(B[i%6])  code segment:  2,4]  y() values)  opy(values) |

9

1

Q11. Identify the error(s) in the following questions

(i) a=[1,2,3,2,1]

for x in range(a): print(x)

* 1. a=["Unnati","Pratham","Utkarsh","Shlok"]

for x in range(len(a)):

print(a(x))

* 1. A=5\*(None)

A=8\*[3]

for i in range(5): A[i]=eval(A[i]\*i)

for i in A:

print(A(i))

1. Assuming that nums is a list of integers, write code to find the length of nums without using the inbuilt function len().
2. Assuming that *num* is a list of integers, write Python code to divide all those elements of list by 5 which are divisible by 5 and multiply the other list elements by 2. The program should display the original list and the changed list. No element of the list should be changed to float as a result of division by 5. The code should display the original list and the changed list.
3. Assuming that num is a list of integers, write code display the list. The code should then replace each element having even value with its half and each element having odd value with its double, and then display the changed list.

Example: if list has: 3, 4, 5, 16, 9, then the modified list should have: 6, 2, 10, 8, 18

1. Write a program to input 10 numbers from the user and then display this list of numbers in reverse order.
2. Write a program to input 10 numbers from the user and find their sum and average. After that display all those numbers (entered by the user) which are greater than the average.
3. Write a program to create a list of 10 random integers in the range 10 to 99, and display this list. Then display all the odd elements of the list.
4. Write a program to create a list of 10 random numbers in the range 10.00 to 99.99, and display this list. Then interchange first half of the list with second half and display the updated list. For example, if the original list is:

[*98.39, 50.84, 42.57, 65.37, 46.78*, **37.67, 16.98, 76.22, 54.24, 44.7**]

then after interchange, the list should become

[**37.67, 16.98, 76.22, 54.24, 44.7**, *98.39, 50.84, 42.57, 65.37, 46.78*]

1. Write a program to create a list of **n** random integers in the range 1 to 25, and display the list. Then exchange the values of first half side elements with the second half side elements of the list. **n** is to be input from the user. Example:

If a list of eight elements has initial contents as: 2,4,1,6,7,9,23,10 The function should rearrange the list as: 7,9,23,10,2,4,1,6

1. Write a program to create a list of 10 random numbers in the range 10.00 to 99.99, and display this list. Then interchange consecutive elements of the list and display the updated list. For example, if the original list is:

[**88.76**, 41.43, **35.9**, 70.06, **88.77**, 53.68, **33.01**, 83.42, **35.29**, 98.22]

then after interchange, the list should become

[41.43, **88.76**, 70.06, **35.9**, 53.68, **88.77**, 83.42, **33.01**, 98.22, **35.29**]

1. Write a program to create a list NUMBERS of ***n*** random integers and display the list. The program should then reverse the contents of the list without slicing the list and without using any second list.

Example: If the list initially contains 2, 15, 3, 14, 7, 9, 19, 6, 1, 10

then after reversal the list should contain 10, 1, 6, 19, 9, 7, 14, 3, 15, 2

1. Write a program to create a list of 10 random integers in the range 100 to 999, and display this list. Then display all the elements of the list which are prime.
2. Write a program to create a list of 20 random integers in the range 1 to 20 and display this list. The program should then shorten the remove by removing all the duplicate elements from the list and display the shortened list. For example, if the original list is:

[16, 15, 20, 6, 9, 12, 18, **16**, 2, 19, **9**, **20**, **20**, 14, 17, **2**, **19**, **12**, 10,

13]

Then the shortened list should be:

[16, 15, 20, 6, 9, 12, 18, 2, 19, 14, 17, 10, 13]

# (Original sequence of elements should be maintained in the shortened list)

1. Write a program to input two lists A and B, and display these lists. Then from list A delete all the elements which are present in list B too. Then display the list A.
2. Write a program to create two lists A and B. Then sort A in ascending order and B in descending order, and merge the contents A and B into a third list C. The resultant list C is required to be in ascending order. Program should not use the + operator to concatenate the lists.
3. Write a program to create two equi-sized lists A and B, and then combine their contents into a third list C, by computing C[i]=2\*A[i]+3\*B[i]; where value i varies from 0 to N-1, N being the size of each list. Then display the lists A, B, and C.
4. Assuming that two lists first and second already exist, write a program to transfer the content of first and second to a third list ALL. The even places in ALL should be filled from list first and odd places of ALL will be filled from list second. Example:

If first is 1,2,3,4,5,6 and second is 10, 20, 30, 40

The resultant ALL is: 10, 1, 20, 2, 30, 3, 40, 4, 5, 6

1. Write a program to input a list from the user and re-position all its elements by shifting each of them to one position before and by shifting the first element to the last position. For example, If the entered list is is [22, 25, 70, 32, 12], then the changed list should be [25, 70, 32, 12, 22]

(Do not use slicing in the program)

1. Write a program to input a list, P, from the user and modify its content in such a way that the elements, which are multiples of 10 swap with the value present in the very next position in the list. For example

If the content of list P is: 91, 50, 54, 22, 30, 54

Then content of list P should become: 91, 54, 50, 22, 54, 30

1. Write a program to input a list from the user and change all the multiples of 5 in the tuple to 5 and rest of the elements as 0. For example, if the list contains:

[55, 43, 20, 16, 39, 90, 83, 40, 48, 25]

Then after executing the program, the list content should be changed as follows:

[5, 0, 5, 0, 0, 5, 0, 5, 0, 5]

1. Write a program to create a list of random integers and replace each even element of the list with the sum of its digits and each odd element with the product of its digits. For example, if the tuple is given as:

[100, 43, 20, 56, 32, 91, 80, 40, 45, 21]

After executing the program, the tuple content should be changed as follows: [1, 12, 2, 11, 5, 9, 8, 4, 20, 2]

1. To input a list, named *Pay*, from the user and modify each element of Pay, as per the following rules:

|  |  |
| --- | --- |
| Existing value of Pay | Pay to be changed to |
| If less than 100000 | Increase existing value by 25% |
| If >= 100000 and <200000 | Increase existing value by 20% |
| If >= 200000 | Increase existing value by 15% |

1. Write a program to create a list names1 to store the names of n1 students of a class. Also create a list names2 to store the names of n2 students of another class. n1, n2, and the names are to be input from the user. Then combine these two lists into a new list and display this new list in the ascending order of names.
2. Write a menu driven program which creates an empty list and gives the following options to the user to perform various operations on a list:
   1. Append an element to the list
   2. Input an element from the user and remove it from the list
   3. Remove all elements from the list
   4. Count the number of occurrences of an element in the list
   5. Sort the list
   6. Reverse the list
   7. Display the list

The program should terminate when the user chooses the option to exit.

1. Write a menu driven program which creates an empty list and gives the following options to the user to perform various operations on the list:
   1. Input an element and a location from the user and insert the element at the entered location in the list.
   2. Input a location from the user and remove the element from that location in the list.
   3. Input an element from the user and find its location in the list.
   4. Display the list in sorted order without sorting the list.
   5. Display the list in reverse order without reversing the list.
   6. Display the list

The program should terminate when the user chooses the option to exit.

1. Assume that nums is a list in Python and ele is an element to be searched for in the list. Write Python code to search for ele in the list nums without using any inbuilt method of list class. (This is equivalent to index() method of *list* class)
2. The roll numbers, names, and marks of ***n*** students of a class have to be stored in a list. Then a list of all the students having marks more than 75 is to be displayed. Write a Python program to do this.
3. Write a program to input a two digit number and display it in words. For example, if the input is **58**, then the output should be **Fifty Eight**. If the number is not in the range 10 to 99, the program should give an error message.
4. 50 students have to take an exam in an examination hall. Roll numbers of these students are from 91041 to 91090. The invigilator in the examination hall has to maintain a list of roll numbers of all the students present in the hall at any moment. For this we have to write a program for the invigilator which will allow him/her to:
   1. Enter roll numbers of students as they enter the hall (one student at a time). While entry:
      1. Only valid roll numbers should be allowed
      2. No duplicate roll numbers should be allowed
   2. Delete the roll number of any student who leaves the hall
   3. Display the number of students present in the hall at any moment
   4. Display the smallest and the largest roll number present in the hall
   5. Display a list of absentees’ roll numbers
   6. Display a sorted list of roll numbers present