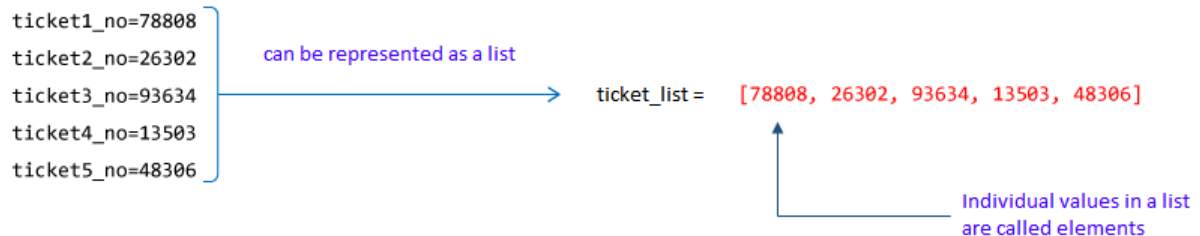


List

List is data type in python which is used to store a group of elements together.

Ex: Suppose we want to store the ticket numbers allocated to each passenger travelling in a flight. Instead of using separate variable for each ticket number, we can use a list as shown below.



Each element in the list has a position in the list known as an index. The list index starts from zero. It's like having seat numbers starting from 0!

Element	78808	26302	93634	13503	48306
Index	0	1	2	3	4

Suppose seat number 2 is allocated to the passenger with ticket number 93634, the passenger can directly go to seat number 2 without having to go through other seats. Similarly, index positions actually help us to **directly access** a value from the list.

Check 1: `list_name[index]` can be used to directly access the list element at the mentioned index position.

Suppose we have to allocate a different passenger to seat number 3, we can do it as `ticket_list[3]=13504`.

Check 2: Thus, in addition to using the index to access an element directly, we can also use it to **directly modify** an element in the list. `ticket_list[2]=99883`

Check 3: we cannot access values beyond the total number of elements in the list.

For example: print `ticket_list[5]` will result in index out of bound error.

`list_of_airlines=["AI", "SJ", "JA", "EM", "AA"]`

Indices of `list_of_airlines`

List of airlines	AI	SJ	JA	EM	AA
Index	0	1	2	3	4
Negative Index	-5	-4	-3	-2	-1

Indices may also be considered negative as shown above. This is normally used to count from right.

Check 4: To fetch the second last airline in the list, we can write `list_of_airlines[-2]`. This is equivalent to `list_of_airlines[len(list_of_airlines)-2]`.

Negative indices can also be used for slicing.

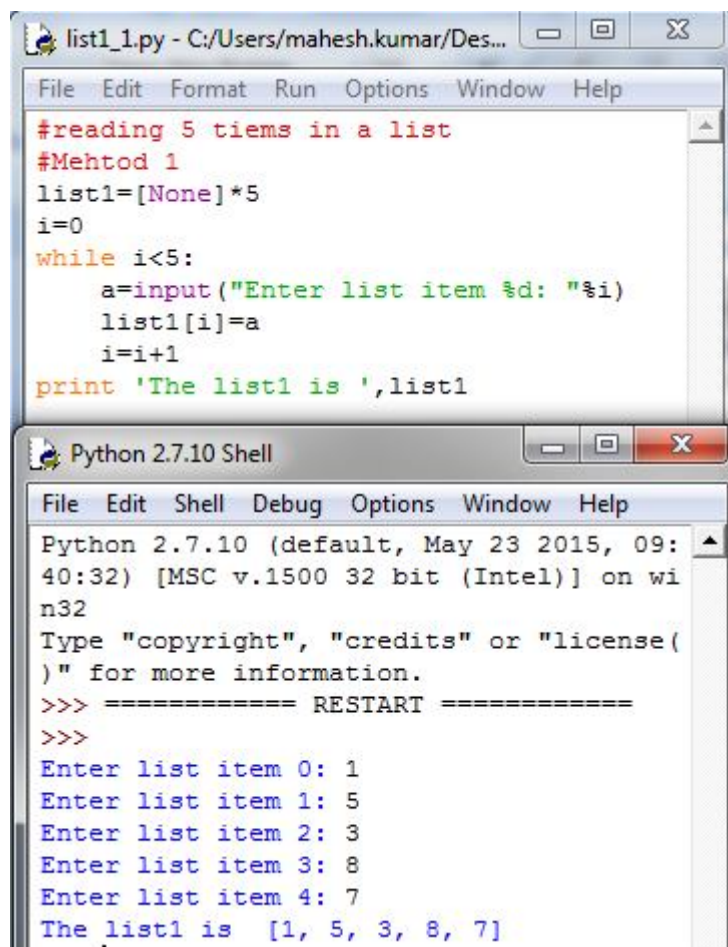
For example: `list_of_airlines[-4:-1]` will give us the same output as `list_of_airlines[1:4]`

It can store heterogeneous data `sample_list=["Mark",5,"Jack",9, "Chan",5]`

Creating a list:

Creating an empty list	<code>sample_list=[]</code>	
Creating a list with known size and known elements	<code>sample_list1=["Mark",5,"Jack",9, "Chan",5]</code> <code>sample_list2=["Mark","Jack", "Chan"]</code>	List can store both homogeneous and heterogeneous elements
Creating a list with known size and unknown elements	<code>sample_list=[None]*5</code>	None denotes an unknown value in Python
Length of the list	<code>len(sample_list)</code>	Displays the number of elements in the list

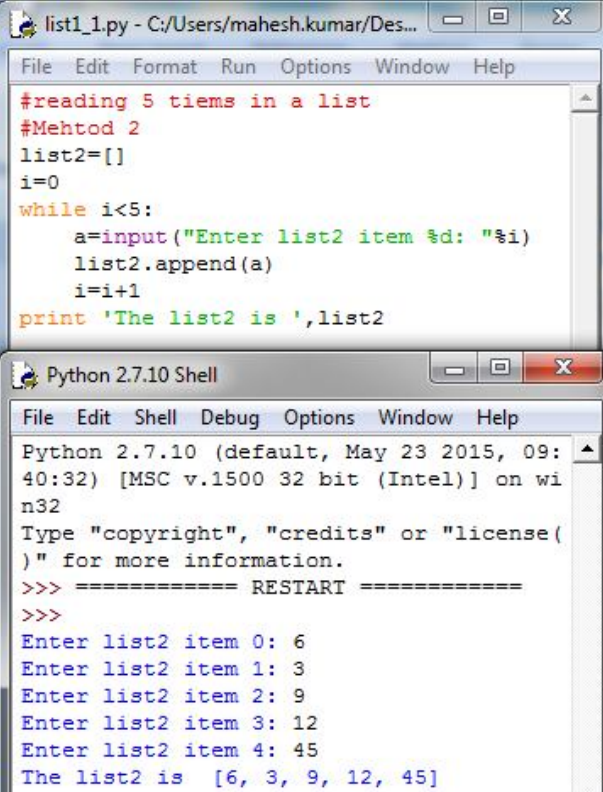
Check 5: input list at run time



```
list1_1.py - C:/Users/maresh.kumar/Des...
File Edit Format Run Options Window Help
#reading 5 tiems in a list
#Mehtod 1
list1=[None]*5
i=0
while i<5:
    a=input("Enter list item %d: "%i)
    list1[i]=a
    i=i+1
print 'The list1 is ',list1

Python 2.7.10 Shell
File Edit Shell Debug Options Window Help
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on wi
n32
Type "copyright", "credits" or "license(
)" for more information.
>>> ===== RESTART =====
>>>
Enter list item 0: 1
Enter list item 1: 5
Enter list item 2: 3
Enter list item 3: 8
Enter list item 4: 7
The list1 is [1, 5, 3, 8, 7]
```

Check 6: input list at run time



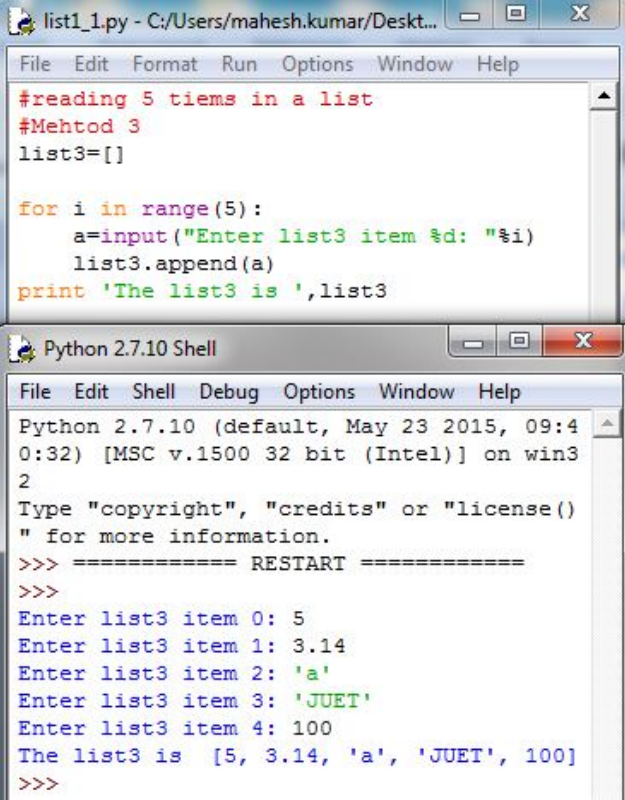
The screenshot shows a Python IDE window titled 'list1_1.py - C:/Users/mahesh.kumar/Des...'. The code in the editor is as follows:

```
#reading 5 tiems in a list
#Mehtod 2
list2=[]
i=0
while i<5:
    a=input("Enter list2 item %d: "%i)
    list2.append(a)
    i=i+1
print 'The list2 is ',list2
```

Below the editor is a 'Python 2.7.10 Shell' window. It displays the output of the script after execution:

```
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
Enter list2 item 0: 6
Enter list2 item 1: 3
Enter list2 item 2: 9
Enter list2 item 3: 12
Enter list2 item 4: 45
The list2 is [6, 3, 9, 12, 45]
```

Check 7: input list using for loop



The screenshot shows a Python IDE window titled 'list1_1.py - C:/Users/mahesh.kumar/Deskt...'. The code in the editor is as follows:

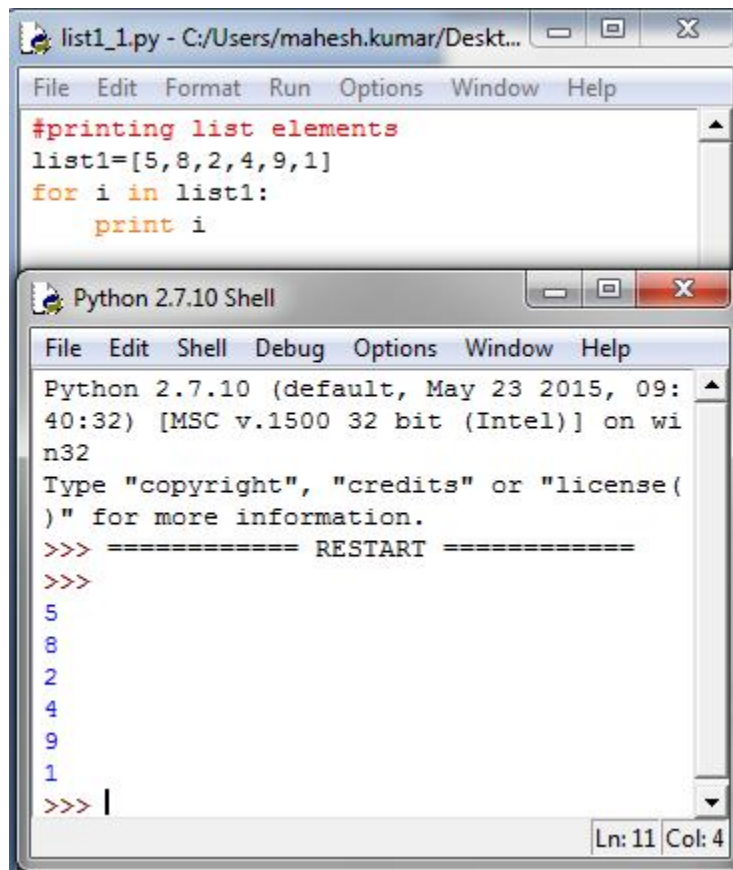
```
#reading 5 tiems in a list
#Mehtod 3
list3=[]

for i in range(5):
    a=input("Enter list3 item %d: "%i)
    list3.append(a)
print 'The list3 is ',list3
```

Below the editor is a 'Python 2.7.10 Shell' window. It displays the output of the script after execution:

```
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
Enter list3 item 0: 5
Enter list3 item 1: 3.14
Enter list3 item 2: 'a'
Enter list3 item 3: 'JUET'
Enter list3 item 4: 100
The list3 is [5, 3.14, 'a', 'JUET', 100]
>>>
```

Check 8: Printing List elements



The screenshot shows a Python IDE window titled 'list1.py - C:/Users/mahesh.kumar/Desktop...'. The code in the editor is:

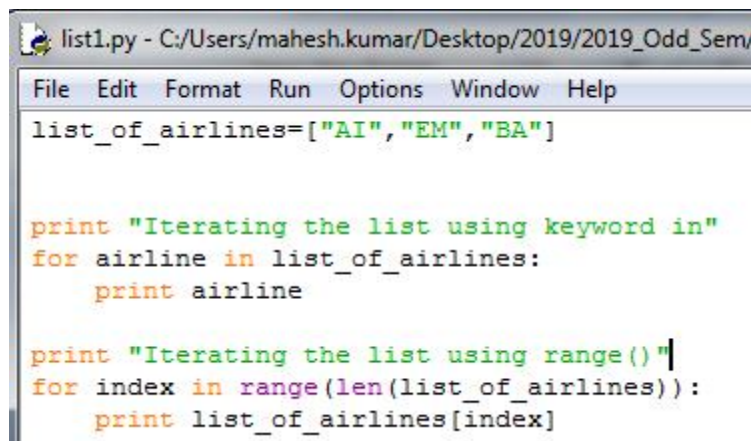
```
#printing list elements
list1=[5,8,2,4,9,1]
for i in list1:
    print i
```

Below the editor is a 'Python 2.7.10 Shell' window showing the execution output:

```
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
5
8
2
4
9
1
>>> |
```

The status bar at the bottom right of the shell window indicates 'Ln: 11 Col: 4'.

Check 9:



The screenshot shows a Python IDE window titled 'list1.py - C:/Users/mahesh.kumar/Desktop/2019/2019_Odd_Sem...'. The code in the editor is:

```
list_of_airlines=["AI", "EM", "BA"]

print "Iterating the list using keyword in"
for airline in list_of_airlines:
    print airline

print "Iterating the list using range()"
for index in range(len(list_of_airlines)):
    print list_of_airlines[index]
```

Prob1: Read elements into a list and show output by printing each elements

Sample Input	Expected Output
1,5,7,8,3.14, 4, 5	1 5 7 8 3.14 4 5
1,[2,3],4,5,6	1 2 3 4 5 6
1,2,[3,4,5],[6,7]	1 2 3 4 5 6 7
[(1,2,3)]	1 2 3

Prob2: Given a list of integer values. Write a python program to check whether it contains same number in adjacent position. Display the count of such adjacent occurrences.

Sample Input	Expected Output
[1,1,5,100,-20,-20,6,0,0]	3
[10,20,30,40,30,20]	0
[1,2,2,3,4,4,4,10]	3

Prob3: Read a list from the user of arbitrary length, and show following:

- print the list entered by the user
- print least value and largest value
- swap positions of least and largest element
- print the list after swapping positions.

Prob4: Read two lists **enrol** and **name** from the user of 10 elements. The list **enrol** contains enrolment numbers and list **name** contains names of the students. Now read enrolment number from the user to search in the list, if the enrolment is found in the list then print enrolment and name of the student. Otherwise print -1.