**2.20 Python List**

The list is an important data type of Python. In a list, we can store many elements under a single variable name. The elements may be of any type. Lists are defined by enclosing the elements inside a pair of square brackets, separated by commas. The individual element may be of any type, even another list.

a=[2,8,14.35, ‘Thomas’]

In the above statement, ‘a’ is called a list in python. It is much more flexible than a string. Using instructions, we can change the order of elements, add new elements, and remove an existing element. This property of the list is called ***mutability***. A list is mutable where a string is immutable. The variable name corresponding to each element in the list is a[0], a[1], a[2] etc. Here 0, 1, 2, 3 etc are called ***index***.

**Example 9:**

*#example for list manipulation*

*a=[3284,'Thomas',81.43]*

*print(a) # Print the list as it is*

*print('roll no ',a[0]) # Print the first element*

*print('Name ',a[1])*

*print('Mark %6.2f'%(a[2])) # formatted printing with list*

*# See how the elements can be changed.*

*a[2]=73.5*

*print('New mark list')*

*print('Mark %6.2f'%(a[2]))*

*#end*

**Output**

*roll no 3284*

*Name Thomas*

*Mark 81.43*

*New mark list*

*Mark 73.50*

In a list, the index of the first element is 0. The statement b=a will not make a new list named ‘b’. Instead of physically copying it will connect b with a as a reference. That means, when we change a, b also will change. Go through the following programme to demonstrate this.

**Example 10:**

*#Example for list manipulation*

*a=[3284]*

*print(len(a)) # To print the number of elements*

*a.append(78.51) # To add a new member at the end*

*a.insert(1,'Antony') # To insert a new element at second position*

*print(a)*

*b=a*

*print(b)*

*a[1]='Thomas' # A correction is made in list ‘a’*

*print(b)*

*#end*

**Output**

*1*

*[3284, 'Antony', 78.51]*

*[3284, 'Antony', 78.51]*

*[3284, 'Thomas', 78.51]*

Please note that when a correction is done in the list a, list b is also changing.

**Table of functions connected with list:**

|  |  |
| --- | --- |
| *list.remove (x)* | Remove the first item from the list whose value is *x*. If there is no such item, the system will report an error. |
| *list.count(x)* | Return the number of times *x* appears in the list. |
| *list.sort()* | Sort the items in the list. |
| *list.index(x)* | Return the index in the list of the first item whose value is *x*. It is an error if there is no such item |
| *list.reverse()* | Reverse the elements of the list |
| *list.append(x)* | Add an item to the end of the list |
| *list.insert(i, x)* | Insert an item at a given position. The first argument is the index of the element before which x is to be inserted. After insertion, the index of the new element becomes i. |
| *len(a)* | Find the number of elements in the list named a. |
| *sum (a,x)* | Sum up all the elements in the list ‘a’ with the value stored in x. |
| *sum(a)* | to add all elements of ‘a’ |
| *x in a* | Test wheatear *x* is a member of a. System returns true if x is a member of ‘a’. |
| *x not in a* | Test wheatear *x* is not a member of a. System returns true, if x is not a member of ‘a’. |
| *min(a)* | If a represents a list, the function will return the element of the lowest value. |
| *del a[x,y]* | Delete all or some of the elements in a list named a. x is the index of the element to remove, and y is the number of elements to remove.  del a[0] to remove the first element  del a[2:4] to remove 4 elements starting from index 2  del a[:] to remove all the elements in the list to make it empty. |
| *sum (a,x)* | Sum up all the elements in the list ‘a’ with the value stored in x. |
| *a+b* | Add list ‘a’ with list ‘b’. |
| *a\*x* | Makes a new list that contains the elements x times by concatenation. |