Programming and Problem Solving through Python Language
O Level / A Level

Chapter - 5: Sequence Data Types

Python Collections (Arrays)

There are four collection data types in the Python programming language:

- **List** is a collection which is ordered and changeable. Allows duplicate members.
- **Tuple** is a collection which is ordered and unchangeable. Allows duplicate members.
- **Set** is a collection which is unordered and unindexed. No duplicate members.
- **Dictionary** is a collection which is unordered, changeable and indexed. No duplicate members.

Lists

- A list is a collection which is ordered and changeable.
- The list is a datatype available in Python which can be written as a list of comma-separated values (items) between square brackets.
- Items in a list need not be of the same type.

Creating List

```
list1 = ['physics', 'chemistry', 1997, 2000];
list2 = [1, 2, 3, 4, 5 ];
list3 = ["a", "b", "c", "d"]
```

Access Items

To access values in lists, use the square brackets for slicing along with the index or indices to obtain value available at that index.

```
list1 = ['physics', 'chemistry', 1997, 2000]
list2 = [1, 2, 3, 4, 5, 6, 7 ]
print ("list1[0]: ", list1[0])
print ("list2[1:5]: ", list2[1:5])
print ("list1[3]: ", list1[-1])
```

Output−

```
list1[0]:  physics
list2[1:5]:  [2, 3, 4, 5]
list1[3]:  2000
```

Negative Indexing

Negative indexing means beginning from the end, -1 refers to the last item, -2 refers to the second last item etc. eg List[-1]

Range of Indexes

- You can specify a range of indexes by specifying where to start and where to end the range.
- **List[2:5]** - The search will start at index 2 (included) and end at index 5 (not included).
- Remember that the first item has index 0.
- **List[:5]** - By leaving out the start value, the range will start at the first item:

Range of Negative Indexes

Specify negative indexes if you want to start the search from the end of the list.

List[-4:-1]

Updating Lists

- You can update single or multiple elements of lists by giving the slice on the left-hand side of the assignment operator, and
- To add an item to the end of the list, use the **append()** method.
- To add an item at the specified index, use the **insert()** method.

```
list = ['physics', 'chemistry', 1997, 2000]
print ("Value available at index 2 : ", list[2])

list[2] = 2001
print ("New value available at index 2 : ", list[2])

list.append('Maths')
print("New value available at index 2 : ", list)


list.append(1,'Hindi')
print("New value available at index 2 : ", list)
```

Loop Through a List

You can loop through the list items by using a for loop:

```
list = ["apple", "banana", "cherry"]
for x in list:
    print(x)
```

Check if Item Exists

To determine if a specified item is present in a list use the "**in**" keyword:

```
list = ["apple", "banana", "cherry"]
if "apple" in list:
    print("Yes, 'apple' is in the fruits list")
```

List Length

To determine how many items a list has, use the **len()** function:

```
print(len(list))
```

Removing Item from the List

- The **remove()** method removes the specified item.

```
list = ["apple", "banana", "cherry"]
list.remove("banana")
```

- The **pop()** method removes the specified index, (or the last item if index is not specified):

```
list.pop()        removes the "cherry"
list.pop(1)       removes the "banana"
```

- The **del** keyword removes the specified index or the complete list.

```
del list[0]
del list
```

- The **clear()** method empties the list

```
list.clear()
```

Basic List Operations

Python Expression	Results	Description
len([1, 2, 3])	3	Length
[1, 2, 3] + [4, 5, 6]	[1, 2, 3, 4, 5, 6]	Concatenation
['Hi!'] * 4	['Hi!', 'Hi!', 'Hi!', 'Hi!']	Repetition
3 in [1, 2, 3]	True	Membership
for x in [1,2,3] : print (x,end = ' ')	1 2 3	Iteration

Built-in Function

Sr.No.	Function & Description
1	len(list) : Gives the total length of the list.
2	max(list) : Returns item from the list with max value.
3	min(list) : Returns item from the list with min value.
4	list(seq): Converts a tuple into list.

List Built-in Methods

Sr.No.	Methods & Description
1	list.append(obj) : Appends object obj to list
2	list.count(obj): Returns count of how many times obj occurs in list
3	list.extend(seq) : Appends the contents of seq to list
4	list.index(obj) : Returns the lowest index in list that obj appears
5	list.insert(index, obj) : Inserts object obj into list at offset index
6	list.pop(obj = list[-1]) : Removes and returns last object or obj from list
7	list.remove(obj) : Removes object obj from list
8	list.reverse() : Reverses objects of list in place
9	list.sort([func]) : Sorts objects of list, use compare func if given

Matrix implementation using list

We can implement matrix operation using list. Matrix operation can be implemented using nested list. List inside another list is called nested list.

Matrix creation

Program -1

```
# A basic code for matrix input from user

R = int(input("Enter the number of rows:"))
C = int(input("Enter the number of columns:"))

# Initialize matrix
matrix = []
print("Enter the entries rowwise:")

# For user input
for i in range(R):          # A for loop for row entries
    a =[]
    for j in range(C):      # A for loop for column entries
         a.append(int(input()))
    matrix.append(a)

# For printing the matrix
for i in range(R):
    for j in range(C):
        print(matrix[i][j], end = " ")
    print()
```

Assignment

1. Define list
2. What is the output of the following code:

   a) print type ([1,2]) b) a= [1, 2, 3, None, ( ), [ ]}
3. Write the output from the following code:

   ```
   A=[2,4,6,8,10]
   L=len(L)
   S=0
   for I in range(1,L,2):
   S+=A[I]
   print “Sum=”,S
   ```
4. For each of the expression below, specify its type and value. If it generates error, write error:
   1. List A= [1, 4, 3, 0]
   2. List B= [„x‟, „z‟, „t‟, „q‟]
   3. List A.sort ( )
   4. List A
   5. List A.insert (0, 100)
   6. List A.remove (3)
   7. List A.append (7)
   8. List A+List B
   9. List B.pop ( )
   10. List A.extend ([4, 1, 6, 3])
5. Create a list that contains the names of 5 students of your class. (Do not ask for input to do so)
   1. Print the list
   2. Ask the user to input one name and append it to the list
   3. Print the list
   4. Ask user to input a number. Print the name that has the number as index (Generate error message if the number provided is more than last index value).
   5. Add “Kamal” and “Sanjana” at the beginning of the list by using „+‟.
   6. Print the list
   7. Ask the user to type a name. Check whether that name is in the list. If exist, delete the name, otherwise append it at the end of the list.
   8. Create a copy of the list in reverse order
   9. Print the original list and the reversed list.
   10. Remove the last element of the list.
6. Write a program to input NXM matrix and find sum of all even numbers in the matrix.
7. Write a program to print upper triangle matrix.
8. Write a program to print lower triangle matrix.
9. Write a program to find sum of rows and columns of the matrix.