



पुर्णा International School

Shree Swaminarayan Gurukul, Zundal

Term – II

Class – XI

Computer Science

Lists

Very Short Answer Type Questions (1 marks each)

Question 1:

How many types of built-in types of sequences in Python.

Answer:

Python has six built-in types of sequences.

Question 2:

Write the output of the given Python code.

```
#!/user/bin/python
list1 = ['physics', 'chemistry', 1997, 2000];
list2 = [1,2,3,4,5,6, 7];
print "list1[0]", list1[0]
```

Answer:

list1[0]: physics

Question 3:

Write the output of the given Python code.

```
#!/user/bin/python
list1 = ['physics', 'chemistry', 1997,2000];
list2 = [1,2, 3,4,5, 6, 7];
print "list2[1:5[ :", list2[1:5]
```

Answer:

list2[1:5[: [2,3,4,5]

Question 4:

How can we remove list element ?

Answer:

To remove a list element, you can use either the del statement or the remove method

Question 5:

Why we use `cmp(list1, list2)` in Python?

Answer:

It compares elements of both lists.

Question 6:

What is the use of `len(list)` in Python?

Answer:

It gives the total length of the list.

Question 7:

Describe `max(list)` method.

Answer:

It returns item from the list with max value.

Question 8:

Describe `min(list)` method.

Answer:

It returns item from the list with min value.

Question 9:

What is the use of `list(seq)` in Python ?

Answer:

It converts a tuple into list.

Question 10:

What do you mean by `list.append(obj)`

Answer:

Appends object `obj` to list

Question 11:

What do you mean by `list.count(obj)`

Answer:

Returns count how many times `obj` occurs in list

Question 12:

Explain `list.extend(seq)`

Answer:

Appends the contents of `seq` to list.

Question 13:

Is `list.reverse()` method return a value?

Answer:

This method does not return any value but reverse the given object from the list

Question 14:

Which method is used to sort objects of list.

Answer:

sort ()

Question 15:

Which function is used to reverse objects of list in place.

Answer:

list.reverse ()

Question 16:

Which function is use for removing object obj from list.

Answer:

list.remove(obj)

Question 17:

Which function is use for returning the lowest index in list that obj appears.

Answer:

List index (obj).

Short Answer Type Questions (2 marks each)

Question 1:

Define list in Python.

Answer:

The list is a most versatile datatype available in Python which can be written as a list of comma- separated values (items) between square brackets. Good thing about a list is that items in a list need not all have the same type. Creating a list is as simple as putting different comma-separated values between square brackets.

For example :

```
list1 = ['physics', 'chemistry', 1997,2000];
```

```
list2 = [1,2, 3,4, 5];
```

```
list3 = ['a', "b", "c", "d"];
```

Question 2:

Give an example to access values in lists.

Answer:

To access value in lists, use the square brackets for slicing along with the index or indices to obtain value available at that index. Following is a simple example :

```
#!/user/bin/python
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]
```

When the above code is executed, it produces the following result:

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

Question 3:

Give an example to update single or multiple elements of lists

Answer:

You can update single or multiple elements of lists by giving the slice on the left-hand side of the assignment operator, and you can add elements in a list with the append() method. Following is a simple example:

```
#!/user/bin/python
listl = ['physics', 'chemistry', 1997, 2000];
print "Value available at index 2 :."
print list[2];
list[2] = 2001;
print "New value available at index 2 :."
print list [2];
Note: append()) method is discussed in subsequent section.
```

When the above code is executed, it produces the following result:

```
Value available at index 2 :
1997
New value available at index 2 :
2001
```

Question 4:

Give an example to remove list element.

Answer:

To remove a list element, you can use either the del statement if you know exactly which ' element(s) you are deleting or the remove () method if you do not know. Following is a simple example:

```
#!/user/bin/python
list1 = ['physics', 'chemistry', 1997, 2000];
print list1;
del list1 [2];
print "After deleting value at index 2 :."
print list1;
```

When the above code is executed, it produces the following result:

```
['physics', 'chemistry', 1997, 2000];
After deleting value at index 2;
['physics', 'chemistry', 2000]
```

Question 5:

Describe list() with its syntax.

Answer:

The method list() takes sequence types and converts them to lists. It is used to convert a given tuple into list.

Syntax

Following is the syntax for list() method :

```
list (seq)
```

Question 6:

Write the output of the given Python code :

```
#!/user/bin/python
list1, list2 = [123, 'xyz'], [456, 'abc']
print cmpt(list1, list2);
print cmp(list2, list1);
list3 = list2 + [786];
print cmp(list2, list3)
```

Answer:

This will produce the following result:

```
-1
1
-1
```

Question 7:

Write the output of the given Python code :

```
#!/user/bin/python
aList = [123, 'xyz', 'zara', 'abc', 123];
bList = [2009, 'manni'];
aList.extend (bList)
print "Extended List :", aList;
```

Answer:

This will produce the following result:

```
Extended List : [123, 'xyz', 'zara', 'abc', 123, 2009, 'manni']
```

Question 8:

Write the output of the given python code :

```
# ! 'user/bin'pvthon
aList1 = [123, 'xvz', zara', abc'];
```

```
print "Index for xyz : " aList. index) 'xyz');  
print "Index for zara :", aList. index('zara');
```

Answer:

This will produce the following result:
Index for xyz : 1 Index for xxx : 2

Question 9:

Write the output of the given python code :
!/user/bin/python
aList = [123, 'xyz', 'zara', 'abc'];
aList.insert (3,2009) print "Final List:", aList

Answer:

Output:
Final List: [123, 'xyz', 'zara', 2009, 'abc']

Question 10:

Write the output of the given python code :

```
#!/user/bin/python  
aList1 = [123, 'xyz', 'zara', 'abc'];  
aList.insert (3,2009) print "Final Lista List
```

Answer:

Output:

Final List: [123, 'xyz', 'zara1, 2009,'abc1']

Question 11:

Write the output of the given python code :

```
# !/user/bin/python  
aList1 = [123, 'xyz', 'zara', 'abc'];  
print "A List:", aList.pop()  
print "B List:", aList.pop(2)
```

Answer:

Output:

A List: abc B List: zara

Question 12:

Write the output of the following code

```
A = [2, 4, 6, 8,10]
L = len (A)
S = 0
for l in range (1, L, 2):
S += A[l]
print "Sum=", S
```

Answer:

```
Sum = 12
Justification:
A[1] = 4 step size = 2
A[3] = 8
S = 4 + 8 = 12
```

Question 13:

How are lists different from strings when both are sequences ?

Answer:

The lists and strings are different in following ways :

- (a) The lists are mutable sequences while strings are immutable.
- (b) Strings store single type of elements, all characters while lists can store elements belonging to different types.
- (c) In consecutive locations, strings store the individual characters while list stores the references of its elements.

Question 14:

Write a program to calculate and display the sum of all the odd numbers in the list.

Answer:

```
pos = 0
sum = 0
while pos < len (L):
if L[pos] %2 == 1 :
sum = sum + L [pos]
pos = pos + 1
print sum
```

Question 15:

Define a function overlapping () that takes two lists and returns True if they have at least one member in common, False otherwise.

Answer:

```
def overlapping (a, b):
l1 = len (a)
l2 = len (b)
for i in range (l1):
```

```
for j in range (12):
if a[i] == b[j] :
return True
else
return False
```

Question 16:

Write a program to find all duplicates in a list.

Answer:

```
a =[1,2,3,2,1,5,6,5,5,5]
d={ }
for elem in a:
if elem in
d: d[elem] + = 1
else:
d[elem] = 1
print elem
print [x for x, y in d.items() if y > 1],
print "have duplicates"
print d
```

Output screenshot:

```
[1,2,5] have duplicates
{1:2,2: 2,3:1,5:4, 6: 1}
```

Long Answer Type Questions (4 marks each)

Question 1:

What is string in Python and how can we access values in list.

Answer:

The most basic data structure in Python is the sequence. Each element of a sequence is assigned a number – its position or index. The first index is zero, the second index is one, and so forth. Python has six built-in types of sequences, but the most common ones are lists and tuples.

There are certain things you can do with all sequence types. These operations include indexing, slicing, adding, multiplying, and checking for membership. In addition, Python has built-in functions for finding the length of a sequence and for finding its largest and smallest elements.

Python Lists:

The list is a most versatile datatype available in Python which can be written as a list of comma-separated values (items) between brackets. Good thing about a list is that items in a list need not all have the same type.

Creating a list is as simple as putting different comma-separated values between square brackets. For example:


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

Like string indices, list indices start at 0, and lists can be sliced, concatenated and so on.

Accessing Values in Lists :

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

```
#!/user/bin/python
list = ['physics', 'chemistry', 1997,2000];
list2 = [1,2, 3,4,5,6, 7];
print "list1 [0]:", list1 [0]
print "list2[1:5]:", list2[1:5]
list 2[1:5]: [2, 3, 4, 5]
```

Question 2:

How can we update and delete list in python.

Answer:

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 you can add to elements in a list with the append() method. Following is a simple example:

```
#!/user/bin/python
list = ['physics', 'chemistry', 1997, 2000];
print "Value available at index 2 :"
```

```
print list[2];
list [2] = 2001;
print "New value available at index 2 :"
```

```
print list [2];
```

When the above code is executed it produces the following result:

```
Value available at index 2 :
1997
```

```
New value available at index 2 :
2001
```

Delete List Elements:

To remove a list element, you can use either the del statement if you know exactly which element(s) you are deleting or the remove() method if you do not know. Following is a simple example :

```
# !/user/bin/python
list1 = ['physics', 'chemistry', 1997, 2000];
print list1; del list1 [2];
print "After deleting value at index 2:"
print list1;
```

When the above code is executed, it produces the following result:

```
['physics', 'chemistry', 1997, 2001]
After deleting value at index 2 :
['physics', 'chemistry', 2000]
```

Question 3:

Write a program to input n x m matrix and find the sum of all numbers.

Answer:

```
def summat (a, m, n):
s = 0
for i in range (m):
for j in range (n):
s += a[i][j]
```

Question 4:

Write a program to pass any list and to arrange all numbers in descending order.

Answer:

```
def arrange (l, n)
for i in range (n - 1):
for j in range (n - i - 1):
if l[j] > l[j + 1] :
temp = l[j]
l[j] = l[j+1]
l[j+1] = temp
```

Question 5:

Write the output for the following codes.

```
A= (10:1000,20:2000,30:3000,40:4000, 50:5000}
print A.items()
print A.keys()
print A.values()
```

Answer:

[(40, 4000), (10, 1000), (20, 2000), (50, 5000), (30, 3000)]

[40,10, 20, 50, 30]

[4000,1000, 2000,5000,3000]