1. What exactly is []?

**Answer:** [] are called Index brackets and are used to define list in python.

For example,

numList =[] #creates an empty list

numList = [1,2,3,4]

[] is also used for retrieving and assigning values from/to a list element based on index.

For example,

numList[2]

>>> 3

numList[1] = 5

>>> assigns the value 5 at 1st index of the list

--------------------------------------------------------------------------------------------------------------------------------------

2. In a list of values stored in a variable called spam, how would you assign the value 'hello' as the third value? (Assume [2, 4, 6, 8, 10] are in spam.)

**Answer:**

spam[2] = ‘hello’

>>>[2,4,’hello’,8,10]

--------------------------------------------------------------------------------------------------------------------------------------

Let's pretend the spam includes the list ['a', 'b', 'c', 'd'] for the next three queries.

3. What is the value of spam[int(int('3' \* 2) / 11)]?

**Answer: d**

--------------------------------------------------------------------------------------------------------------------------------------

4. What is the value of spam[-1]?

**Answer: d**

--------------------------------------------------------------------------------------------------------------------------------------

.5. What is the value of spam[:2]?

**Answer: [‘a’, ‘b’]**

--------------------------------------------------------------------------------------------------------------------------------------

Let's pretend bacon has the list [3.14, 'cat,' 11, 'cat,' True] for the next three questions.

6. What is the value of bacon.index('cat')?

**Answer: 1**

--------------------------------------------------------------------------------------------------------------------------------------

7. How does bacon.append(99) change the look of the list value in bacon?

**Answer: [3.14, 'cat', 11, 'cat', True, 99]**

--------------------------------------------------------------------------------------------------------------------------------------

8. How does bacon.remove('cat') change the look of the list in bacon?

**Answer: [3.14, 11, 'cat', True, 99]**

--------------------------------------------------------------------------------------------------------------------------------------

9. What are the list concatenation and list replication operators?

**Answer:** List concatenation operator is **+** and list replication operator is **\***

--------------------------------------------------------------------------------------------------------------------------------------

10. What is difference between the list methods append() and insert()?

**Answer:** The append() method adds the element to the end of the list and the insert() method will insert the element at a specific index

For example,

N1 = [1,2,3,4]

N1.append(5) #appends 5 at last

>>>[1,2,3,4,5]

N1.insert(1,6) #inserts 6 at 1st index

>>>[1,6,2,3,4,5]

--------------------------------------------------------------------------------------------------------------------------------------

11. What are the two methods for removing items from a list?

**Answer:**

1. Remove(): removes the element from the first occurrence of it.
2. Pop(): removes the last element of the list

--------------------------------------------------------------------------------------------------------------------------------------

12. Describe how list values and string values are identical.

**Answer:** The similarity between list values and string values is both are sequences. The len() method can be used in both, both are used in loops, they have indexes and slices, both can be concatenated and replicated.

However, the difference are lists are mutable whereas strings are immutable and all the characters of a string is of type string whereas a list can have multiple type elements in it.

--------------------------------------------------------------------------------------------------------------------------------------

13. What's the difference between tuples and lists?

**Answer:** Mutability is the difference. Tuples are immutable whereas lists are mutable. List is defined in [] and tuple is defined in ()

--------------------------------------------------------------------------------------------------------------------------------------

14. How do you type a tuple value that only contains the integer 42?

**Answer:**

To create a tuple with only one element, we need to add a comma(,) at the end

For example,

tuple1 = (42,)

#The values can be accessed using the index

Tuple[0]

>>>42

--------------------------------------------------------------------------------------------------------------------------------------

15. How do you get a list value's tuple form? How do you get a tuple value's list form?

**Answer:** By type converting/casting.

For example,

List1 = [1,2,3,4] #list

>>>[1,2,3,4]

Tuple1 = tuple(List1) #converts list to tuple

>>>(1,2,3,4)

List2 = list(Tuple1) #converts tuple to list

>>>[1,2,3,4]

--------------------------------------------------------------------------------------------------------------------------------------

16. Variables that "contain" list values are not necessarily lists themselves. Instead, what do they contain?

**Answer:** The contain the references to the list values

--------------------------------------------------------------------------------------------------------------------------------------

17. How do you distinguish between copy.copy() and copy.deepcopy()?

**Answer:** copy.copy() creates a shallow copy of the list i.e., modifying the copied list modifies the original list as well whereas copy.deepcopy() creates a new object and copies the list and modifying the copied list will not affect the original list.