1. What exactly is []?

Ans-It’s a empty list , which is a list value that contains no items.

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.)

Ans-spam[2]=’hello’

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)]?

Ans- output will be ‘d’.

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

Ans-output will be ‘d’.

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

Ans- ['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')?

Ans-1

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

Ans-bacon will update as [3.14, 'cat', 11, 'cat', True, 99]

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

Ans-bacon will update as [3.14, 11, 'cat', True, 99], only first cat will be removed.

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

Ans- The operator for list concatenation is +, while the operator for replication is \*.

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

Ans- The difference between the two methods is that . append() adds an item to the end of a list, whereas . insert() inserts and item in a specified /index in the list.

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

Ans-The methods are remove(), pop() and clear(). remove() helps to remove the very first given element matching from the list. The pop() method removes an element from the list based on the index given. The clear() method will remove all the elements present in the list.

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

Ans-Both strings and lists have lengths: a string's length is the number of characters in the string; a list's length is the number of items in the list. Each character in a string as well as each item in a list has a position, also called an index.

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

Ans-The primary difference between tuples and lists is that tuples are immutable as opposed to lists which are mutable. Therefore, it is possible to change a list but not a tuple. The contents of a tuple cannot change once they have been created in Python due to the immutability of tuples.

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

Ans- (42,) (The trailing comma is mandatory.)

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

Ans-Using tuple(list\_name). Typecasting to tuple can be done by simply using tuple(list\_name). To convert a tuple into list in Python, call list() builtin function and pass the tuple as argument to the function. list() returns a new list generated from the items of the given tuple

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

Ans-Variables will contain references to list values rather than list values themselves. But for strings and integer values, variables simply contain the string or integer value.

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

Ans-

The copy() returns a shallow copy of the list, and deepcopy() returns a deep copy of the list. As you can see that both have the same value but have different IDs.

The difference between shallow and deep copying is only relevant for compound objects (objects that contain other objects, like lists or class instances):

* A shallow copy constructs a new compound object and then (to the extent possible) inserts references into it to the objects found in the original.
* A deep copy constructs a new compound object and then, recursively, inserts copies into it of the objects found in the original.