1. What exactly is []?

**Ans:-** It’s a list. A list is a data type which can hold values of different data types inside it means in a 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.)

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

**Ans:- 1.)**spam = [2, 4, 6, 8, 10]

spam[2] = 'hello'

print(spam)

2.) spam = ['a', 'b', 'c', 'd']

spam[2] = 'hello'

print(spam)

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

**Ans:-** The value of following in the list of ['a', 'b', 'c', 'd'] is ‘d’ and the value of following argument in the list of [2, 4, 6, 8, 10] is 8.

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

**Ans:-** The value of following argument in the list of ['a', 'b', 'c', 'd'] is ‘d’ and the value of it in the list of [2, 4, 6, 8, 10] is 10.

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

**Ans:-** The value of following argument in the list of ['a', 'b', 'c', 'd'] is [‘a’, ‘b’] and the value of it in the list of [2, 4, 6, 8, 10] is [2,4].

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:-** The value of following argument and the index of ‘cat’ in the following list is 1.

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

**Ans:-** When we use append method to add some value in a list it always adds this value as a last element in the list, so after we append this value inside a list then the list will look like this [3.14, 'cat', 11, 'cat', True, 99].

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

**Ans:-** The remove function is used to remove the content of a list, but there is one disadvantage of this function means if you want to remove ‘cat’ from this above list but if you have more than one values as a ‘cat’ in that particular list then it will only remove the first occurrence of that value ‘cat’ in the above list, so after running the above function the list will like [3.14, 11, 'cat', True]. See it has only removed the first occurrence of the value ‘cat’ from the list.

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

**Ans:-** The list concentenation operator is ‘+’ operator which is used to add two strings, values, numbers, lists, etc and the list replication operator is asterisk symbol ‘\*’ .

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

**Ans:-** Append(), and insert() both the function are used to add new elements in the list but there is one difference between both of these functions. When you add a elements in a list using a append method, the append method take only one value as input which will be the value which you want to add into that list and the append() method always adds up the value as a last element in the list, but if you use insert() function to add new elements in a list, it takes two inputs, the first input will be the index number means in which index number you want this value to get added in that particular list, you can provide that index number and the value will get added in that provided index number and and the second input will be the value which you want to add in that particular list, and the element will only get added in your specified index in that particular list.

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

**Ans:-** The remove(), and pop() function both are used to remove elements from a list but there is one difference between both of them, When we use remove() function to remove a contents from a list , it will only remove the first occurrence of that particular element from the list if the value which you want to remove from a list have more than one occurrences in the list. but the pop() method is used to remove the contents of a list using its index value, when we use pop() method, we need to provide the index number of that element in a particular list and it will remove the element from that index from the list.

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

**Ans:-** The items in a list and string both have have a specific order that specifies the way, how you can access it. You can also access individual items in a list or string using index value. We can also access the subset of a list or string by providing a range of index values. Both strings and lists supports the len() function, which is used to determine the length of a list or string. Both lists and strings use the ‘+’ operator to be concantenated. We can two or more lists or strings using ‘+’ operator.

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

**Ans:-** Lists are mutable means you can add, remove and modify items in a list but on the other hand Tuples are immutable means you can’t add, remove and modify contents inside the tuple. We create a list using [] square brackets while on the other hand we use () brackets to create a tuple. Lists have many built-in methods, that can be used to modify contents in the list because they are mutable but on the other hand tuples have fewer built-in methods because they are immutable.

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

**Ans:-** To create a tuple which contains the only integer 42, we can use the parentheses to enclose the value.

Tuple = (42,)

The comma after the integer is necessary to indicate that this is a tuple with only one element otherwise python would interpret the value as simple integer rather than a tuple if we don’t use comma inside it.

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

**Ans:-** To get a list values in the form of tuple we can use the python’s built-in function ‘Tuple()’, and to get the tuple values in the form of a list we can use the ‘list()’ built-in function of python.

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

**Ans:-** In python, Variables that contain list values are not necessarily list themselves, but rather they can contain the references to the list objects. When we create a list and assign it to any variable, the variable does not actually contain the list values directly, rather it contains the reference to the list objects that is stored in the memory.

This means that we can assign the same list to the multiple variables, and if we change the list present inside any of the variables, the changes would get reflected in all the lists which are present inside other variables.

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

**Ans:-** When we use copy.copy(), it creates a copy of the original object with the same elements as the original object but if we make any changes in the original object it will get reflected in the copied object as well but on the other hand, if we use copy.deepcopy(), it also creates a copy of the original object with the same elements as the original object but in this method if we make any changes in the original object, it will not get reflected in the copied object.

In Python inside copy module, copy() and deepcopy() are the two methods which is used to create a shallow and deep copy of an object.