1. What exactly is []?

These square brackets are known as index brackets, it used to define lists and indexing.

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.)

spam= [2, 4, 6, 8, 10]

spam.insert(2, 'hello')

print(spam)

**Output:** [2,4,’hello’,6,8,10]

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)]?

spam= ['a', 'b', 'c', 'd']

a=spam[int(int('3' \* 2) / 11)]

print(a)

**Output:** d

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

spam= ['a', 'b', 'c', 'd']

a=spam[-1]

print(a)

**Output:** d

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

spam= ['a', 'b', 'c', 'd']

a=spam[:2]

print(a)

**Output:** [‘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')?

bacon=[3.14, 'cat', 11, 'cat', True]

a=bacon.index('cat')

print(a)

**Output:** 1

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

bacon=[3.14, 'cat', 11, 'cat', True]

bacon.append(99)

print(bacon)

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

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

bacon=[3.14, 'cat', 11, 'cat', True]

bacon.remove('cat')

print(bacon)

**Output:** [3.14, 11, 'cat', True]

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

The list concatenation and list replication operators are ‘+’ and ‘\*’.

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

Append() method is used to add the values to the list, but it adds the value at the end of the list.

Insert() method is used to add the values to the list, but it adds the value at the desired position(index).

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

To remove items for list two methods are: remove() and clear().

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

Values in lists and string are identical as they are ordered.

Both lists and strings can be passed to len() function, have indexes and slices, be used in for loops, be concatenated or replicated, and be used with the in and not in operators.

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

Tuples are not mutable whereas lists are mutable.

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

It will be written as (42,) where the trailing comma is mandatory otherwise it’ll be considered as integer.

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

The tuple() and list() functions, respectively are used to convert a list to tuple and vice versa

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

They contain references to the list value.

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

Copy.copy() (shallow copy) stores the copy of the original object and points the references to the objects. It doesn't create a copy of nested objects; instead it just copies the references to the nested objects

Copy.deepcopy() (Deep copy) stores the copy of the original object and recursively copies the objects as well. It copies all the nested objects recursively. a, b, c, d, a1, b1, c1 and d1 are references to objects in memory, which are uniquely identified by their ids.