1. What exactly is []?

ans- empty 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.)

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 – ‘d’

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

Ans - 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 - [3.14, 'cat', 11, 'cat', True,99]

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

Ans - [3.14, 'cat', 11,True,99]

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

Ans - + , \*

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

Ans – appens() will insert a new element at the end of a list. Insert() will insert a new element at the specified index given by the user.

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

Ans – remove() and pop()

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

Ans – string and list are ordered sequential collection. String method works similar in list and string.

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

Ans – tuples are immutable datatype means once it defined we cannot change. But lists are mutable datatype.

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

Ans – tu = (42,)

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

Ans – if **tu** is a tuple then its list value is **list(tu)**.

If **a** is a list then tuple value is tuple(a).

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

Ans - string

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

Ans – in copy.copy() any changes in the nested objects in old list, then that changes show as in new list . but in copy.deepcopy() the above mentioned operation do not reflect.