1. What exactly is []?

Ans :- It is a symbolic representation of 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 = [2,4,6,8,10]

Spam.insert (3, “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, 11, 'cat', True, 99]

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

Ans :- list concatenation operator is +

List replication operators is \*

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

Ans :- append use to add data at the last of list

Insert use to insert data in list at particular index

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

Ans :- list.pop()

List.remove()

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

Ans :- Both are sequences

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

Ans :- The main difference between tuples and lists is that tuples are immutable object whereas lists are mutable objects.

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

Ans :- (42)

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

Ans :- tuple = ( [3,2,4,3,], [6,3,2,5], [6,8,4,5,7] )

list = [(2,4,1,2), (6,7,8,9), (2,4,1,3)]

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

Ans :-

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

Ans :- copy.copy() in this method any change made to a copy object do reflect in original object

copy.deepcopy() in this method any change made to a copy object do not reflect in original object.