1. What exactly is []?

* 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.)

* 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]?

* [‘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')?

* 1

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

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

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

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

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

* List concatenation -> +
* list replication -> \*

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

* append() -> inserts new value to the last
* insert() -> inserts new value at given index

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

* .remove(), .pop()

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

* Both list and string stores heterogenous data types. Indexing works in list as well as in strings

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

* Lists are mutable and tuples are immutable.
* Lists supports insertion and deletion, tuples is read-only
* Lists have more inbuilt methods, whereas tuples have less
* Lists takes more memory than tuples and more time consuming than tuples.

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

* (42, )

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

* Using list() -> converts tuple to list
* using tuple() -> converts list to tuple

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

* List is a heterogenous data type. It can store any data string, tuples, dictionary, int, float....

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

* copy.copy() -> create new copy but have references to old data, if old data is modified, new copy is also modified but if some data is newly appended to old data, it wont reflect in new copy because it doesnt have references to newly appended data
* copy.deepcopy() -> it will create a new copy which looks similar to old data. But if old data is modified, new copy doesnt reflect.