**1. What exactly is []?**

Answer 1: [] represents an empty list in Python.

**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.)**

Answer 2: 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)]?**

Answer 3: ‘d’

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

Answer 4: ‘d’

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

Answer 5: [‘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')?**

Answer 6: The index position will be 1

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

Answer 7: The element will get added in the list. New list will be: [3.14, 'cat', 11, 'cat', True, 99]

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

Answer 8: It will remove the element. New list will be: [3.14, 11, 'cat', True, 99]

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

Answer 9: The list concatenation operator is ‘+’ and the list replication operator is ‘\*’.

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

Answer 10: The append() method adds an element to the end of the list, while the insert() method inserts an element at a specified index.

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

Answer 11: The two methods for removing items from a list are remove() and pop(). The remove() method removes the first occurrence of a specified value, and pop() removes an element at a specified index (or the last element if no index is specified) and returns the removed value.

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

Answer 12: Both list values and string values are ordered sequences of elements. Each element in a list or string can be accessed by an index, and both support various operations like slicing and iteration.

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

Answer 13: Tuples and lists are both ordered collections, but the key difference is that lists are mutable (can be changed after creation), while tuples are immutable (cannot be changed after creation).

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

Answer 14: my\_tuple = (42)

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

Answer 15:

To convert a list to a tuple, you can use the tuple() function:

list\_value = [1, 2, 3]

tuple\_form = tuple(list\_value)

To convert a tuple to a list, you can use the list() function:

tuple\_value = (4, 5, 6)

list\_form = list(tuple\_value)

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

Answer 16: Variables that "contain" list values store references to the list objects, not the lists themselves.

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

Answer 17: copy.copy() performs a shallow copy, creating a new object but not creating copies of the nested objects. copy.deepcopy()performs a deep copy, creating a new object and recursively copying all the nested objects.