1. What exactly is []?

**Answer:**

[] is used to create a 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.)

**Answer:**

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

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:**

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

value of spam[int(int('3' \* 2) / 11)] is ‘d’

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

**Answer:**

The value of spam[-1] is ‘d’

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

**Answer:**

The value of spam[:2] is [‘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:**

The value of bacon.index(‘cat’) is 1

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

**Answer:**

bacon.append(99)

now bacon list value changed to [3.14, 'cat', 11, 'cat', True,99]

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

**Answer:**

bacon.remove('cat')

New look of bacon list is

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

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

**Answer:**

List concatenation operator is +

List replication operator is \*

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

**Answer:**

List append() method append an element at the end of the list

List insert() method insert an element at the index given.

l1.append(80) => [3.14, 11, 'cat', True, 80]

l1.insert(1,90) => [3.14, 90,11, 'cat', True, 90]

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

**Answer:**

l1.remove()

l1.pop()

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

**Answer:**

String is a sequence of letters in a specific order. Indexing and slicing are used to grab elements of string.

List is a sequence of objects in a specific order. Indexing and slicing is used to grab elements of list.

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

**Answer:**

Tuple is immutable and list is mutable.

List is defined by enclosing its elements in square brackets (**[]**) .

Tuple is defined by enclosing its elements in parentheses (**()**).

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

**Answer:**

t= (42,)

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

**Answer:**

bacon\_l = [3.14, 'cat', 11, 'cat', True, 99]

bacon\_t = tuple(bacon\_l)

print("new tuple is ",bacon\_t)

li = list(bacon\_t)

print(li)

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

**Answer:**

Variables that “contain” list values are not necessarily lists themselves. Instead they contain strings.

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

**Answer:**

Copy.copy() returns the shallow copy of objects .

Copy.deepcopy() returns the deep copy of objects.

Ex :

l1=[1,2,[4,5],6,7]

l2 = copy.copy(l1)

l3 = copy.deepCopy(l1)

copy.copy(l1) returns the shallow copy of list l1. Shallow copy constructs new collection object and inserts the **references** into it to the objects found in the original.

Copy.deepCopy(l1) returns the deep copy of list l2. Deep copy constructs new collection object and inserts the **copies** into it to the objects found in the original.