1. What exactly is []?

This is the 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, 4, 6, 8, 10]

spam [2]='hello'---------this will replace the third value with ‘hello’

print(spam)

Or

spam= [2, 4, 6, 8, 10]

spam.insert(2,'hello') ---this will add ‘hello’ as the third value.

print(spam)

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)]?

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

print(spam[int(int('3' \* 2) / 11)])

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

‘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] because in python indexing starts form zero(0)

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

bacon=[3.14, 'cat', 11, 'cat', True]

bacon.append(99)

print(bacon)

it will append 99 to the list bacon.

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

bacon=[3.14, 'cat', 11, 'cat', True]

bacon.remove('cat')

print(bacon)

It will remove fist ‘cat’ form the list, because remove method take first element as the argument.

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

List concatenation is to add the other data type to list by using ‘+’

List replication is to replicate the other date type to the list by using ‘\*’

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

Append method append the element to list, and takes element as the argument as well append the element to the end to the list.

Insert method takes index and the element both as the argument, so we can decide at which index we want to insert the element.

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

pop () ---this will pop the index and this take index as the argument.

remove () ---this is remove the element and it take element as the argument.

clear () ---this will clear the list.

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

List values and string values are identical.

* They both are mutable
* They both allows indexing

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

Lists are mutable data structure where tuples are immutable data structure.

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

a=(42,)

print(type(a))---this will give us tuple

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

**From tuple to list**

a=(43,)

print(type(a))

a=list(a)

print(type(a))

**From list to tuple**

a=[43]

print(type(a))

a=tuple(a)

print(type(a))

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

They can be any data type with any data structure.

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

**List\_1= [1, 2, 3, 4, 5]**

**List\_2=list\_1.copy ()**

Copy.copy changes the values of the both list when we perform nested list task. For example

**List\_1= [1, 2, 3, 4], [5, 6, 7, 8]**

**List\_2=list\_1.copy ()**

And when we do any changes in nested list like above list\_1 has two elements and if we do any changes inside the objects of that elements, then both the list will change.

**List\_2 [1] [1] = 23**

Then this value will change in both of the list.

Unlike copy.copy () in copy.deepcopy () the value of the both list will not change.

**List\_2 [1] [1] = 23**

This will only change the value of list\_2