1. What exactly is []?\

Answer : used to write expressions that evaluate to a single item within a list, or a single character in a string. Also, used to pass an 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.)

Let's pretend the spam includes the list ['a', 'b', 'c', 'd'] for the next three queries.

Answer: snap = [2, 4, 6, 8, 10]  
snap[2] = 'Hello'  
print(snap)

3. What is the value of spam[int(int('3' \* 2) / 11)]?

Answer : 3 because ‘3’\*2 is 33 since it is in ‘int’.

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

Answer: span = [343,45,45,2,3]  
x= span[-1]  
print(x) #output = 3

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

Let's pretend bacon has the list [3.14, 'cat,' 11, 'cat,' True] for the next three questions.

Answer: span = [343,45,45,2,3]  
x= span[:2]  
print(x) #output = 343,45

span = [3.14, 'cat,' 11, 'cat,' True]  
x= span[:2]  
print(x) #output = 343,45

Output :[3.14, 'cat,']

6. What is the value of bacon.index('cat')?

Answer : Output: [3.14, 'cat']

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

Answer : None

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

Answer : [3.14, 11, 'cat,', True]

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

Answer: Concatenation:

s1=**"Welcome "**s2=**"to "**s3=**"python"**s4=s1+s2+s3  
print (s4)*#Output:Welcome to python*

Replication : l1=[1,2,3]

print (l1 \* 3)

#Output:[1, 2, 3, 1, 2, 3, 1, 2, 3]

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

Answer: Append -

list = [1,2,3,4,5]

list.append(6)

print(list) # [1,2,3,4,5,6]

Use of Insert:

list = [1,2,3,4,5]

list.insert(5, 10) # [1,2,3,4,5,10]

list.insert(1, 10) # [1,10,3,4,5]

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

Answer: Remove() It helps to remove the very first given element matching from the list.

pop() The pop() method removes an element from the list based on the index given.

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

Answer: The similarity between Lists and Strings in Python is that both are sequences

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

Answer: tuples are immutable (that is you cannot change the items that a tuple contains, and you can’t change the order of the items it contains).

lists are mutable (that is you can change the items that a list contains, and you can change the ordering of the contained items).

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

Answer: (42,)

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

Answer: Using tuple built in function

sample\_list = ['Compile', 'With', 'Favtutor']

tuple1 = tuple(sample\_list)

print(tuple1)

print(type(tuple1))

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

Answer: the container object only hold references (pointers) to the stored values.

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

ANSWER: DEEP COPY

old\_list = [[1, 1, 1], [2, 2, 2], [3, 3, 3]]

new\_list = copy.deepcopy(old\_list)

print("Old list:", old\_list)

print("New list:", new\_list)

Ouput: Old list: [[1, 1, 1], [2, 2, 2], [3, 3, 3]]

New list: [[1, 1, 1], [2, 2, 2], [3, 3, 3]]

Copy.copy()

old\_list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]

new\_list = copy.copy(old\_list)

print("Old list:", old\_list)

print("New list:", new\_list)

Output: **Old list: [[1, 2, 3], [4, 5, 6], [7, 8, 9]]**

**New list: [[1, 2, 3], [4, 5, 6], [7, 8, 9]]**