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

This is list data structure where we can store the data. It is similar as array but not same. It is mutable and indexed data. But It can store heterogeneous data in single list like int, float, string where as in array they can store only homogenous data whether int or string or float or complex etc.

1. **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'

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

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

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

print(spam[int(int('3' \* 2) / 11)])

d

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

d

1. **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.**

Corrected [3.14, 'cat’, 11, 'cat’, True]

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

1

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

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

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

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

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

**List Concatenation operation:** It is the operation of list by you can add the two and more list to make a single list. Here you will make new list.

Syntax: New\_list=list\_1+list\_2

**List Replication operation:** Replication is an operation by you can extend the list by addition of another list in main list, which list you want to extend. But here we are not making any new list just doing extension of list by extend operation.

Syntax: List\_1.extend(list\_2)

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

**Append () :** By this method you can add the element in last of the list.

Syntax : list\_1.append(element)

**Insert():** You can also add the element by this method but This method you have choice, You can add the element at anywhere in the list on the basis of index.

Syntax : list\_1.insert(index, value)

This method is slower than append if you don’t add at last of the list, on the basis of O Notion method.

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

**a.Pop():** You Can delete the last element in the list.

Syntax : list.pop()

**b. remove():**  You can remove any element by the name of element.

Syntax : list.remove(element)

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

String and list are almost same nature wise String and list both can be iterable by the index.

String and list are mutable its element can replace by the any other element.

String and list can be editable on the basis of the desired requirement like delete but not be added in string any extra any element.

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

Tuple is faster than list.

Tuple immutable whether list is mutable.

In Tuple cannot be add any other element as requirement, it is fixed data structure can be not be change.

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

(42,)

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

Conversion from list to tuple

tuple(['a','b','c'])

Conversion from tuple to list

list(('a','b','c'))

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

Type(a='b',[1,3,4])

Tuple

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

Copy.copy: This is called shallow copy. It’s create the new variable and refer to original object.

When you change in clone or copied object then change will also happen in real object as well.

Deep.copy : It creates the original copy of the refer to original object. When you change in copy clone then no change will happen in original object.