1. What exactly is []?

Ans:Empty list

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.)

Ans: spam.insert(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)]?

Ans: ’d’

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

Ans: ‘d’

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

Ans:[‘a,’b’]

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

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

Ans:1

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

ANS:[3.14,’cat’,11,’cat’,True,99]

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

Ans:[3.14,’cat’,11,True,99]

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

Ans: list concatenation will combine the elements of the lists in a single list and the list replication will replicate the values inside the list as many time as it is instructed to replicate by the integer value provided.

Example:

List1=[3,”hello”,7,9.7]

List2=[8,”star”,9.6,8]

List+List2 will give [3,”hello”,7,9.7,8,”star”,9.6,8]

List1\*3 will give [3,”hello”,7,9.7,3,”hello”,7,9.7,3,”hello”,7,9.7]

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

Ans: Append will insert the value in the end position of the list shifting all elements to the left and insert will insert the value in the mention location of the list shifting all the elements located at that position of the original list to the right of the list.

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

Ans:remove and pop

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

Ans: list values and string values both can be access by indexes and slicing, also we can run a loop for all the values in the list and similarly we can run a loop for all the characters in the string.

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

Ans:lists are mutable but tuples are immutable.

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

Ans:let us say that the tuple is stored in a variable named as U1.

U1=(42,)

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

Ans: Let t1 be variable declaring a tuple and l1 be a variable declaring a list. Then by Typecasting we can get the rewuired form from tuple value or list value. Just use list(t1) and

tuple(l1) for getting list form of tuple value and list value’s tuple form respectively.

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

Ans:They contain reference or pointer to the memory location where the list is stored.

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

Ans: copy.copy(x) return a shallow copy of x but copy.deepcopy(x) return deep copy of x.

However, copy ( shallow copy) doesn't create a copy of nested objects, instead it just copies the references to the nested objects, while deepcopy ( deep copy) copies all the nested objects recursively. Some examples to demonstrate the behaviour of copy and deepcopy: