**Ans: 1. '[]' this is square bracket and we use it to represent list datatype and it means emply list.**

**Ans: 2. spam = [2, 4, 6, 8, 10]**

**spam[3] = "Hello"**

**Ans: 3. spam = ['a', 'b', 'c', 'd']**

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

**Ans: 4. The value of spam[-1] is : d**

**Ans: 5. The value of spam[:2] is : ['a', 'b']**

**Ans: 6. bacon = [3.14, 'cat,' 11, 'cat,' True]**

**The value of bacon.index('cat') is : 1**

**As index() returns the index of first matching object.**

**Ans: 7. bacon.append(99) results : [3.14, 'cat', 11, 'cat', True, 99]**

**As append() function adds object to the last of a list[].**

**Ans: 8. bacon.remove('cat') results : [3.14, 11, 'cat', True]**

**As the remove() function removes the exact matching objects.**

**Ans: 9. The list concatenation operator is '+' and list replication operator is '\*'**

**As '+' operator takes both argument type of operands as list and results list**

**And '\*' operator takes one operand argument as list and other as int & results list.**

**Ans: 10. list append() adds the object at last index of list by default**

**and insert() adds the element at a particular index**

**eg:-**

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

**spam.append(100)**

**spam.insert(3, 'iNeuron')**

**print(spam)**

**['a', 'b', 'c', 'iNeuron', 'd', 100]**

**Ans: 11. The two methods for removing items from a list are :- remove() & pop()**

**Ans: 12. The common features between Lists and Strings in Python is that both are sequences, for both indexing slicing is applicable.**

**The differences between them are that**

**(1) Lists are mutable but Strings are immutable**

**(2) list can store heterogeneous objects whereas string can store only string type.**

**Ans: 13. List: Tuple:**

**i. list is mutable. | i. tuple is immutable.**

**ii. We represent list by [] and brackets are mandatory. | ii. We represent tuple by () and brackets are optional.**

**iii. Packing and unpacking are not applicable. | iii. Packing and unpacking concepts are applicable.**

**iv. To store list elements PVM requires more memory. | iv. To store tuple elements PVM requires less memory.**

**v. We can access list elements lower as performance | v. We can access tuple elements faster as performance is less.**

**Ans: 14. t = (42,)**

**Ans: 15. t = (1,2,3,4,5)**

**l = list(t)**

**l = ['a','b','c','d','e']**

**t = tuple(l)**

**Ans: 16. Variables that "contain" list values are not necessarily lists themselves. Instead, they can contain the fundamental datatypes i.e:- int, float, double, str & bool**

**Ans: 17. copy() creates a new reference to original object. If you perform any changes to copied object it reflects in original object.**

**deepcopy() creates new object same as original object. Changes performed in new deepcopied object doesn't affect the original object.**