1. What exactly is []?

* []:- Square bracket is use to create a list. A list is created by placing elements inside [].

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.)

* List = [2,4,6,8,10]
* List[2]=”Hello”
* List=[2,4,”Hello”,8,10]

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[int(int(“3”\*2)/11)]
* Spam[int(int(33)/11)]
* Spam[int(int(3)]
* Spam[3]
* Output = ‘d’

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

* Output=’d’, Negative Indexes count from the end.

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

* Output=[‘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')?

* Output= 1

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

* Output= [3.14, 'cat', 11, 'cat', True, 99]

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

* Output= [3.14, 11, 'cat', True, 99]

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

* The operator for list concatenation is +, while the operator for replication is \*. (This is the same as for strings.)
* A=[1,2,3]
* B=[4,5,6]
* C=A+B (Concatenation) and C=A\*2 (Replication)
* C=[1,2,3,4,5,6] and C=[1,2,3,1,2,3]

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

* While append() will add values only to the end of a list, insert() can add them anywhere in the list.

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

* List.pop()
* List.remove()
* List.clear()

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

* Strings and lists are similar. One simple difference between strings and lists is that lists can any type of data i.e. integers, characters, strings etc, while strings can only hold a set of characters.

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

* The tuples are immutable objects the lists are mutable.

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

* Tuple= (42,) 🡪 the trailing comma is mandatory.

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

* Data=[1,2,3,4,5]
* Data\_tuple=tuple(Data)
* Type(data\_tuple) 🡪 Output = Tuple
* Data\_list=list(Data)
* Type(data\_list)=List

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

* X={"k1":1,"k2":[2,4,5,6,3],"k3":(54,568,2462,24825)}
* type(X) 🡪 Output= dict
* Y= (1,2,[1,2,3,4,5],5,(5,6,7,8),10)
* type(Y) 🡪 Output=tuple

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

* Deepcopy: - In copy.deepcopy() we can copy list1 into list2. If we change value in list2 that will not reflect list1 that is called deepcopy.
* Any changes made to a copy of object do not reflect in the original object.
  + Import copy
  + List1=[1,2,3]
  + List2=copy.deepcopy(list1)
  + List2[2]=4
  + print(list1) 🡪 [1,2,3]
  + print(list2) 🡪 [1,2,4]
* Shallow copy: - In copy.copy() we can copy list1 into list2. If we change value in list2 that will reflect in list1 that is called deepcopy.
  + Import copy
  + List1=[1,2,3]
  + List2=copy.copy(list1)
  + List2[2]=4
  + print(list1) 🡪 [1,2,4]
  + print(list2) 🡪 [1,2,4]