**Name: Shahiraj B Lakade**

1. What exactly is []?

**Ans:-[] is a 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[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 value of spam is 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, ,' 11, 'cat,' True]**

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

**Ans concatenation(+) is used to add data object and replication(\*) is used to multiply it**

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

**Ans In append() we append object at end while insert() helps us to put object in specific index**

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 they both are ordered collections of characters**

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

**Ans list are mutable and tuples are immuntable**

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

**Ans we can write for eg t=(42)**

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

**Ans we can write t=tuple(list) ; a=list(tuples)**

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

**Ans They contain index number**

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

**Ans The copy.copy() function will do a shallow copy of a list, while the copy.deepcopy() function will do a deep copy of a list. That is, only copy.deepcopy() will duplicate any lists inside the list.**