PYTHON BASIC

Assignment_4

1. What exactly is []?

The empty list value, which is a list value that contains no items.

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.)

spam[2] = 'hello'

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

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

spam[3] >>d

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

>> d

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

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

>> 11

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

>> 1

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

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

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

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

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

The operator for list concatenation is +, while
the operator for replication is *

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

append() :add always at the last position (.append)**,while**
insert() : add at the given position (.insert(position_index,replacement))

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

>> pop()
>> remove()

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

The similarity between Lists and Strings is that both are sequences.

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

- **Lists are mutable but Strings are immutable.**
- **Secondly, elements of a list can be of different types whereas a String only contains characters that are all of String type.**

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

```
t=tuple([42])
t
```

>>42

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

```
t=tuple([42])
t
```

```
type(t)
```

Tuple

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

- **deepcopy means first constructing a new collection object and then recursively populating it with copies of the child objects found in the original.**
- **In case of deep copy, a copy of an object is copied to another object. It means that any changes made to a copy of an object do not reflect in the original object.**

- **A shallow copy creates a new object which stores the reference of the original elements.So, a shallow copy doesn't create a copy of nested objects, instead it just copies the reference of nested objects. This means, a copy process does not recurse or create copies of nested objects itself.**
- **The changes made in the copied object also reflect the original object.**
- **Shallow copy is faster than Deep copy.**
- **It stores references of the object in the main memory.**