assignment4

June 5, 2023

1. What exactly is [ ]?

ANS:The empty list value, which is a list value that conatins no item.This is similar to how ' ' is the empty string value.like a =[ ].

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.

[1]:
```
# solution by changing the value in index 3
spam = [2, 4, 6, 8, 10]
spam[2] = 'hello'
spam
```

[1]:
```
[2, 4, 'hello', 8, 10]
```

[4]:
```
# solution by inserting value in 3rd index
spam = [2, 4, 6, 8, 10]
spam.insert(2,'hello')
spam
```

[4]:
```
[2, 4, 'hello', 6, 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)]?

[10]:
```
spam = ['a', 'b','c','d']
spam[int(int('3' * 2) / 11)]     # spam[int(33/11)] = spam[3]
```

[10]:
```
'd'
```

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

[18]:
```
spam = ['a', 'b','c','d']
spam[-1] # negative indexes count from the end.
```

[18]:
```
'd'
```

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

[20]:
```
spam =['a', 'b','c','d']
spam[:2]  #the index starts from 0, so :2 means the number till 2 index, which
 ↪means the alphabet c is in 2 index.
```

[20]:
```
['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)?

[11]:
```
bacon =  [3.14, 'cat', 11, 'cat', True]
bacon.index('cat')       # it returns the index of first occurrence of 'cat'
```

[11]:
```
1
```

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

[17]:
```
bacon =  [3.14, 'cat', 11, 'cat', True]
bacon.append(99)     # append adds the item at the end of the list
bacon
```

[17]:
```
[3.14, 'cat', 11, 'cat', True, 99]
```

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

[19]:
```
bacon =  [3.14, 'cat', 11, 'cat', True]
bacon.remove('cat')    # remove first occurrence of item
bacon
```

[19]:
```
[3.14, 11, 'cat', True]
```

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

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

[21]:
```
l1 = [1,3]
l2 = [7,9]
# list concatination
l1+l2
```

[21]:
```
[1, 3, 7, 9]
```

[25]:
```
l1 = [1,3,4]
# list replication
l1*4
```

[25]:
```
[1, 3, 4, 1, 3, 4, 1, 3, 4, 1, 3, 4]
```

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

ANS: append( ) ->Appends object to the end of the list

insert( ) -> Insert object before index

[30]:
```
bacon =  [3.14, 'VIPUL','DAKSH',  True]
bacon.append('rohit')     # append adds the item at the end of the list
bacon
```

[30]:
```
[3.14, 'VIPUL', 'DAKSH', True, 'rohit']
```

[31]:
```
bacon =  [3.14, 'VIPUL','DAKSH',  True]
bacon.insert(2,'rohit')     # Insert object before index
bacon
```

[31]:
```
[3.14, 'VIPUL', 'rohit', 'DAKSH', True]
```

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

ANS:The pop ( ) and the remove ( ) list methods are two ways to remove values from list.

remove(item) - removes first occurence of a item

pop() - Remove and returns item at index (default last).

[33]:
```
bacon = [3.14, 'cat', 11, 'cat', True]
bacon.remove('cat')
bacon
```

[33]:
```
[3.14, 11, 'cat', True]
```

[34]:
```
bacon = [3.14, 'cat', 11, 'cat', True]
bacon.pop()
bacon
```

[34]:
```
[3.14, 'cat', 11, 'cat']
```

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

ANS:1. Both lists and strings can be passed to len()

2. Have indexes and slices
3. Can be used in for loops
4. Can be concatenated or replicated
5. Can be used with the in and not in operators

13. What is the difference between tuples and lists?

ANS:Lists :

are mutable - they can have values added, removed, or changed.

lists use the square brackets, [and]

Tuples :

are immutable; they cannot be changed at all.

Tuples are written using parentheses, (and) while

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

[44]:
```
tup = (42,)
tup
```

[44]:
```
(42,)
```

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

[46]:
```
#By using tuple() and list() functions
l1 = [2,3]
l = tuple(l1)
l
```

[46]:
```
(2, 3)
```

[48]:
```
fast = [6,9]
l = list(fast)
l
```

[48]:
```
[6, 9]
```

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

ANS: They contain references to list value.

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

ANS:The copy.Copy( ) function will do a shallow copy of a list, The copy.deepcopy ( ) function will do a deep copy of a list. That is only copy.deepcopy( ) will duplicate any lists inside the list.

[ ]:
```

```