

1. What exactly is []?

Answer: list

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.)

Answer: spam[2] = 'hello'

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)]?

Answer: 'd'

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

Answer: 'd'

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

Answer: ['a', 'b']

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

Error in input: bacon = [3.14, 'cat', 11, 'cat', True]

Corrected input: bacon = [3.14, 'cat', 11, 'cat', True]

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

Answer: 1

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

Answer: [3.14, 'cat', 11, 'cat', True, 99]

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

Answer: [3.14, 11, 'cat', True, 99]

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

Answer:

concatenation -> +

replication -> *

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

Answer:

`append()` -> is used to add an element in end of the list.

`insert()` -> is used to insert element in specific index.

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

Answer:

`del bacon[-1]`

`bacon.remove(99)`

`bacon.pop()`

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

Answer:

-> Strings are immutable sequences of Unicode code points.

-> Lists are mutable sequences, typically used to store collections of homogeneous items.

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

Answer:

tuple -> Immutable (can not be changed)

list -> Mutable (can be modified)

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

Answer: a = tuple(42) or a = (42)

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

Answer:

Use type casting for it.

Example:

```
tuple([1,2,3,4]) # list to tuple
```

```
list((1,2,3,4)) # tuple to list
```

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

Answer:

Variable contains the reference of the list. It may contain heterogeneous element like string, integer, float etc.,.

Example:

```
a = ['a', 1, 2.3]
```

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

Answer:

copy.copy() method creates a copy of an existing list. The copy() method is added to the end of a list object and so it does not accept any parameters.

copy.deepcopy() is a process in which the copying process occurs recursively. It means that any changes made to a copy of object do not reflect in the original object.
