

```
In [1]: ##1. what exactly is []?
```

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

File "<ipython-input-1-3b186d643f6d>", line 3
 The empty list value, which is a list value that contains no items. This is similar to how '' is the empty string value.
 ^
SyntaxError: invalid syntax

```
In [2]: ## 2.in a list of variables stored in a variable called spam, how would assign the
## assume[2,4,6,8,10] are in spam.
```

```
spam[2] = 'hello' ##(Notice that the third value in a list is at index 2 because the
```

NameError Traceback (most recent call last)
 <ipython-input-2-ccb61fe534d8> in <module>
 2 ## assume[2,4,6,8,10] are in spam.
 3
 ----> 4 spam[2] = 'hello' ##(Notice that the third value in a list is at index 2 because the first index is 0.)
NameError: name 'spam' is not defined

```
In [3]: ## 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)]?
```

'd' ##(Note that '3' * 2 is the string '33', which is passed to int() before being divided by 11.
 This eventually evaluates to 3. Expressions can be used wherever values are used)

File "<ipython-input-3-c64f9edfb618>", line 4
 'd' (Note that '3' * 2 is the string '33', which is passed to int() before being divided by 11.
 ^
SyntaxError: invalid syntax

```
In [4]: ## 4. what is the value of spam[-1]?
```

'd' ##(Negative indexes count from the end.)

Out[4]: 'd'

```
In [6]: ## 5. what is the value of spam[:2]?
```

['a', 'b']

Out[6]: ['a', 'b']

```
In [7]: ##Let's pretend bacon has the list [3.14,"cat",11,"cat",True]
## 6.what is the value of [3.14,'cat','11','cat',True]
```

Object `generations` not found.

```
In [26]: bacon=[3.14,'cat',11,'cat',True]
a= bacon.index('cat')
print(a)
```

1

In [13]: *## 7.how does bacon.append(99) change the look of the list value in bacon?*

In [30]: `bacon=[3.14,'cat',11,'cat',True]
bacon.append('99')
bacon`

Out[30]: [3.14, 'cat', 11, 'cat', True, '99']

In [15]: *## 8.how does bacon.append(99) change the look of the list value in bacon?*

In [31]: `bacon=[3.14,'cat',11,'cat',True]
bacon.remove('cat')
bacon`

Out[31]: [3.14, 11, 'cat', True]

In [17]: *## 9. what are list concatenation and list replication operators?
The operator for list concatenation is +, while the operator for replication is *.*

File "<ipython-input-17-71ae5b093f4f>", line 2
The operator for list concatenation is +, while the operator for replication is
*.
^
SyntaxError: invalid syntax

In [18]: *## 10.what is the difference between list method append() and insert()?
While append() will add values only to the end of a list, insert() can add them anyw*

File "<ipython-input-18-b7febd852c22>", line 3
While append() will add values only to the end of a list, insert() can add them
anywhere in the list.
^
SyntaxError: invalid syntax

In [19]: *## 11.what are two methods for removing items from a list?
The del statement and the remove() list method are two ways to remove values from a*

File "<ipython-input-19-583cf581f568>", line 3
The del statement and the remove() list method are two ways to remove values fro
m a list.
^
SyntaxError: invalid syntax

In [20]: *## 12.Describe how string values and list values are identical?
Both lists and strings can be passed to len(), have indexes and slices, be used in f
and be used with the in and not in operators.*

File "<ipython-input-20-e73e798a2931>", line 3
Both lists and strings can be passed to len(), have indexes and slices, be used
in for loops, be concatenated or replicated,

^

SyntaxError: invalid syntax

In [21]: *## 13. difference between tuple and list?*

Lists are mutable; they can have values added, removed, **or** changed. Tuples are immutable. Also, tuples are written using parentheses, (**and** **)**, **while** lists use the square brackets.

File "<ipython-input-21-ff4956737e49>", line 3
 Lists are mutable; they can have values added, removed, or changed. Tuples are immutable; they cannot be changed at all.

^

SyntaxError: invalid syntax

In [22]: *## 14. how do you type a value that contains only a integer 42?*

(42,)

Out[22]: (42,)

In [23]: *## 15. how do you get a list value's tuple form? how do you get a tuple value's list form?*

The tuple() **and** list() functions, respectively

File "<ipython-input-23-01f744b0191f>", line 3
 The tuple() and list() functions, respectively

^

SyntaxError: invalid syntax

In [24]: *## 16. variables that "contain" list values are not necessarily lists themselves . I*

They contain references to list values.

File "<ipython-input-24-4315e9b14853>", line 3
 They contain references to list values.

^

SyntaxError: invalid syntax

In []: *## 17. How do you distinguish between copy.copy() and copy.deepcopy()?*

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