Put True (T) or False (F) in every cell, based on characteristics of each type.

1

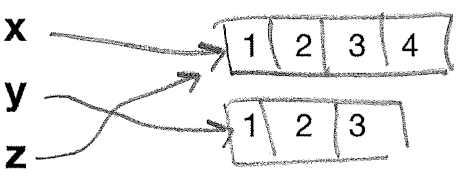
Data Type	Mutable?	Pre-installed?	Builtin?	Create New Types?	Named Attributes?
list	T	T	T	F	F
tuple	F	T	T	F	F
namedtuple	F	F	T	T	T

must import

(done for you)

2

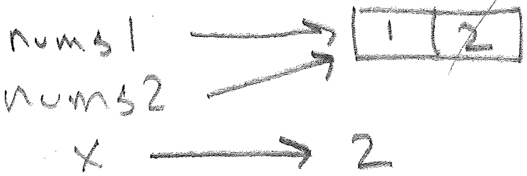
```
x = [1, 2, 3]
y = [1, 2, 3]
z = x
z.append(4)
```



(draw)

3

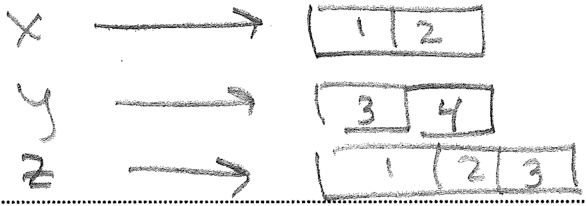
```
nums1 = [1,2]
nums2 = nums1
x = nums2.pop(1)
```



(draw)

4

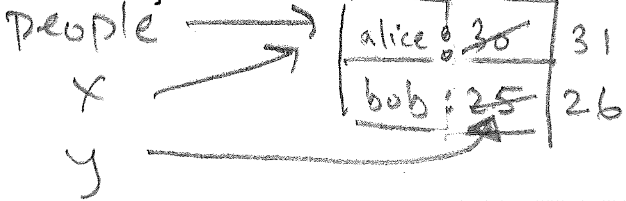
```
x = [1, 2]
y = [3]
z = x + y
y.append(4)
```



(draw)

5

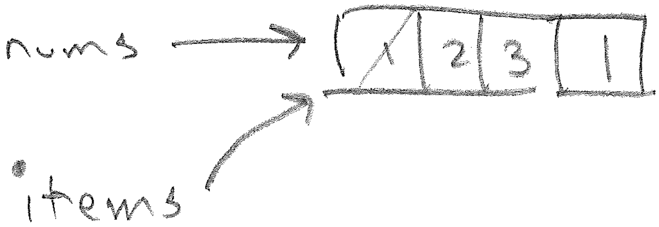
```
people = {"alice":30, "bob":25}
x = people
y = people["bob"]
x["alice"] = 31
y = 26
```



(draw)

6

```
def f(items):
    return items.pop(0)
nums = [1,2,3]
nums.append(f(nums))
```

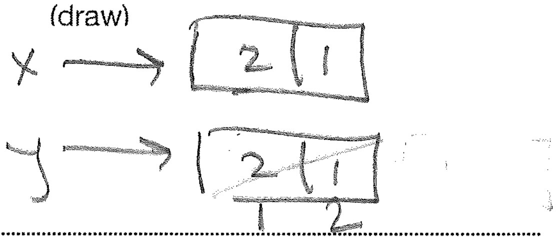


Remember to import copy for these in Python Tutor!

(draw)

7

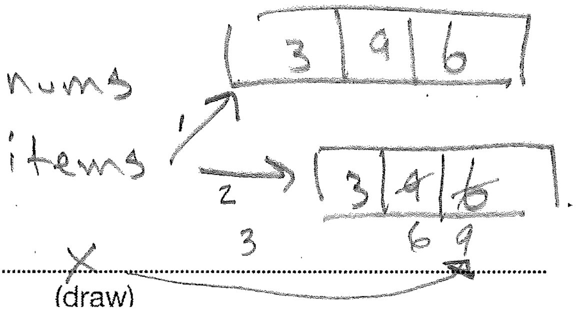
```
x = [2,1]
y = copy.copy(y)
y.sort()
```



(draw)

8

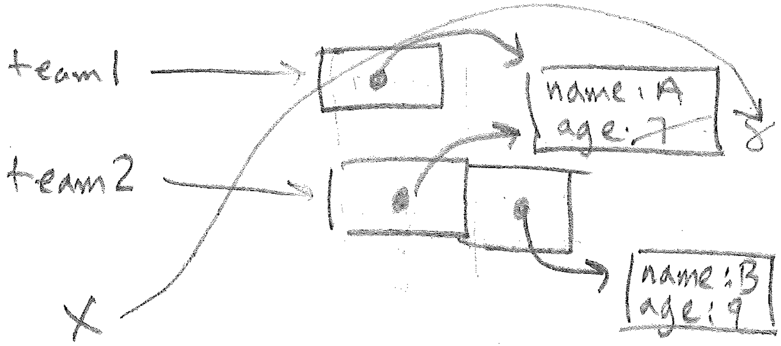
```
line
1 def biggest(items):
2     items = copy.copy(items)
3     items.sort()
4     return items[-1] 9
5 nums = [3,9,6]
6 x = biggest(nums)
```



(draw)

9

```
team1 = [
  {"name":"A", "age":7}
]
team2 = copy.copy(team1)
team2.append(
  {"name":"B", "age":9}
)
team2[0]["age"] = 8
x = team1[0]["age"]
```

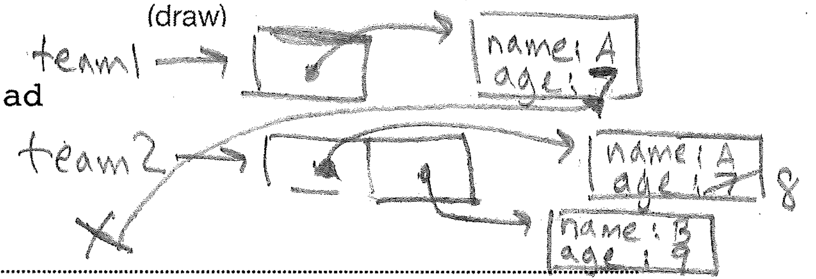


(draw)

10

Same as above, but with copy.deepcopy(...) instead of copy.copy(...).

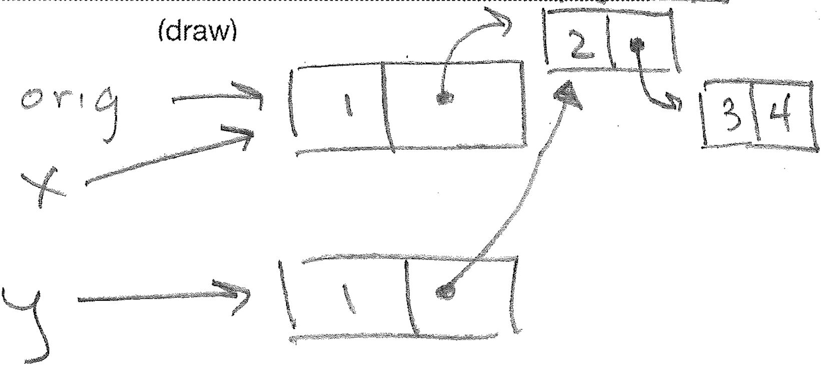
x = 7



(draw)

11

```
orig = [1,[2,[3,4]]]
x = orig
y = copy.copy(orig)
z = copy.deepcopy(orig)
```



orig
1
2
3
4
x
y
1