1) Warm-up: references and aliasing

For each of the following programs, draw a reference diagram and determine the program’s output.

a) 
```python
odds = [1, 3, 5]
odds[0] = 7
nums = [7, 3, 5]
print(odds, nums)
print(odds == nums)
print(odds is nums)
```

b) 
```python
tens = [10, 20, 30]
nums = tens
nums[2] = 40
print(nums, tens)
print(nums == tens)
print(nums is tens)
tenslice = tens[:]
tenslice[1] = 50
print(tenslice, tens)
print(tenslice == tens)
print(tenslice is tens)
```

c) 
```python
first = ['a', 'b']
second = ['c']
first.append(second)
second.append(['d'])
print(first, second)
print(first[2] == second)
print(first[2] is second)
first.append(['e'])
third = ['d']
print(first, third)
print(first[3] == third)
print(first[3] is third)
```

d) 
```python
def myfunc(b):
    b[0] = 2
    return b

a = [1, 3, 5]
c = myfunc(a)
print(a, c)
print(a is c)
```
Tuples Example

def stats(nums):
    total = 0
    for num in nums:
        total += num
    cnt = len(nums)
    avg = total / cnt
    return (cnt, avg)

(count, average) = stats([1,2,3,4,5])
print("Count: " + str(count))
print("Average: " + str(average))

Output:
Count: 5
Average: 3.0

2) Programming practice
   a) Write a function called unique_characters that returns a list of every distinct character that appears in a string. For example, unique_characters("mississippi") returns ['m', 'i', 's', 'p'].

b) Write a function called double that takes a list of numbers and doubles each number in the list.

c) Write a function called double_preserve that takes a list of numbers and returns a list of numbers where each number in the original is doubled. The list passed as a parameter must remain unchanged!