As you walk in

Quiz will start at the beginning of lecture

- Have pencil/pen ready
- Don’t use your own scratch paper
  - We have some if you need it
- Silence phones

Quiz

Before we start

- Don’t open until we start
- Make sure your name and Andrew ID are on the front
- Read instruction page
- No questions (unless clarification on English)

Additional info

- 30 min

15-112
Lecture 2

Week 4 Tue
1-D Lists

Instructor: Pat Virtue

Post Quiz Fun

How many rectangles are drawn for r=100?

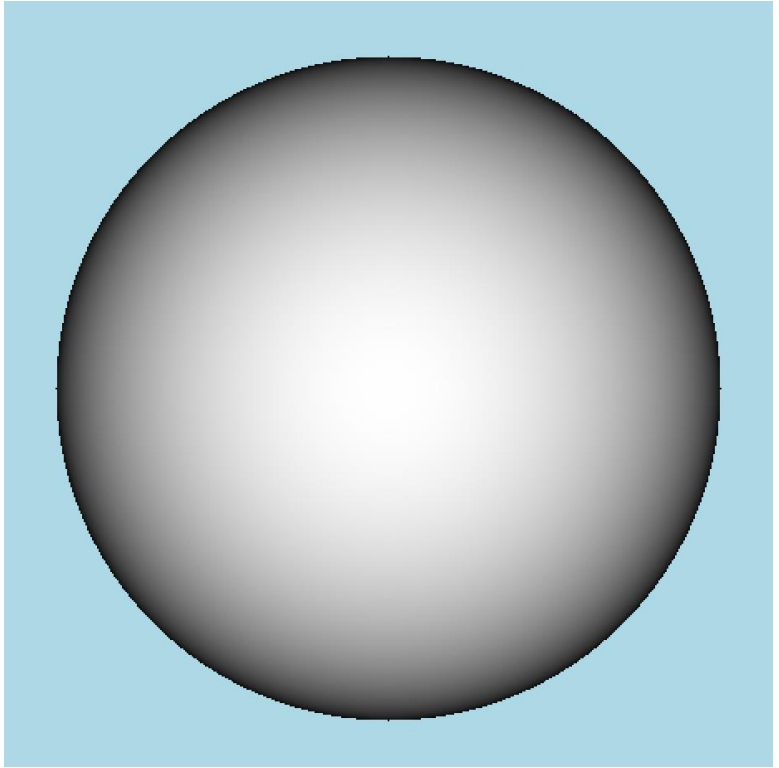
```
def drawSphere(canvas, cx, cy, r):
  for x in range(cx-r, cx+r+1):
    for y in range(cy-r, cy+r+1):
      xSphere = x - cx
      ySphere = y - cy

      # 2D radius
      rxy = math.sqrt(xSphere**2 + ySphere**2)

      if rxy <= r: # Inside 2D radius
        zSphere = math.sqrt(r**2 - xSphere**2 - ySphere**2)

        zGray = int(zSphere/r * 255)
        color = rgbString(zGray, zGray, zGray)

        canvas.create_rectangle(x, y, x+1,y+1,
            fill=color, outline='')
```



Circle math:

$x^2 + y^2 = r$

Sphere math:

$x^2 + y^2 + z^2 = r$

Announcements

Assignments

Week 5 Pre-reading Checkpoint

- Checkpoint out Wed
- Due Fri 9/23, 8 pm

HW4

- Out this evening
- Due Saturday 9/24, 8 pm
- Points will be deducted for style going forward

Quiz

Week 4 material

- Tue 9/27, in lecture

Lists

Poll 1

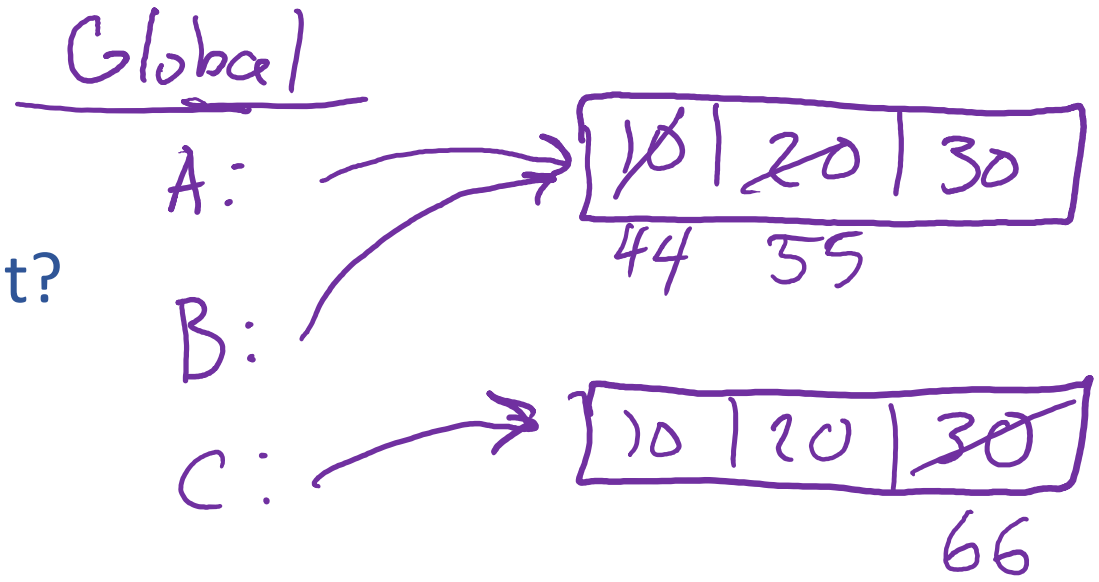
What does this print?

```
import copy

A = [10, 20, 30]
B = A
C = copy.copy(A)

A[0] = 44
B[1] = 55
C[2] = 66

print('A:', A)
print('B:', B)
print('C:', C)
```



I. A: [44, 20, 30]
   B: [10, 55, 30]
   C: [10, 20, 66]

II. A: [44, 55, 30]
   B: [44, 55, 30]
   C: [10, 20, 66] ←

III. A: [44, 20, 66]
   B: [10, 55, 30]
   C: [44, 20, 66]

IV. A: [44, 55, 66]
   B: [44, 55, 66]
   C: [44, 55, 66]

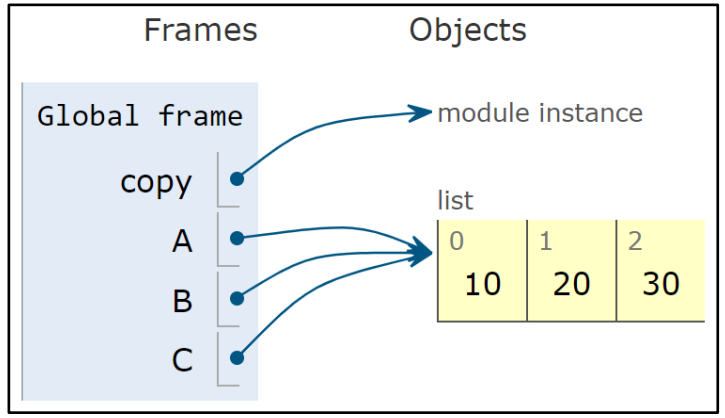
Poll 2

Which is the correct visualization?

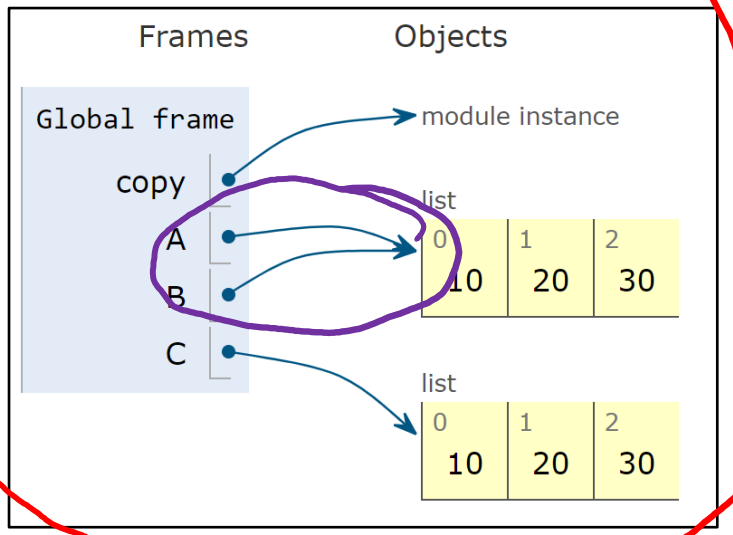
```
import copy

A = [10, 20, 30]
B = A
C = copy.copy(A)
```

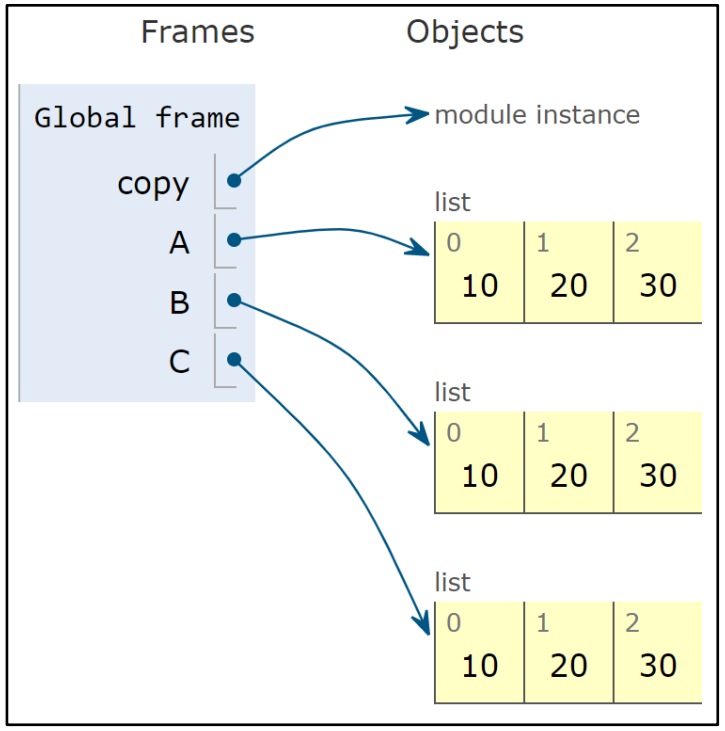
I.



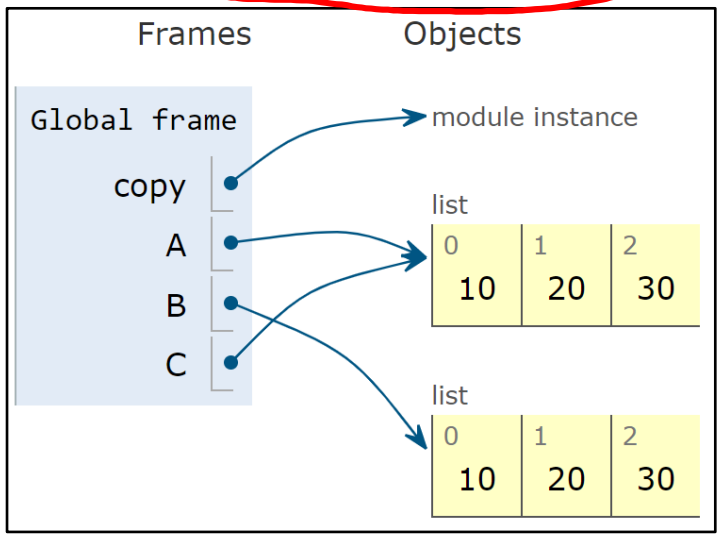
II.



III.



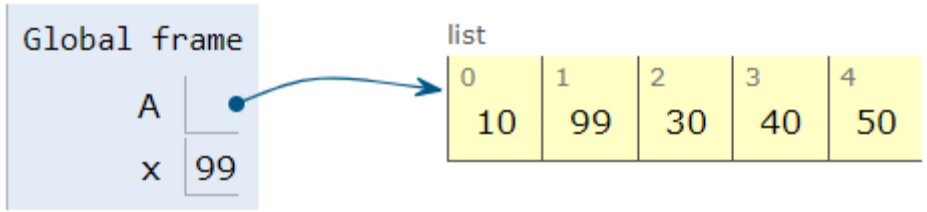
IV.



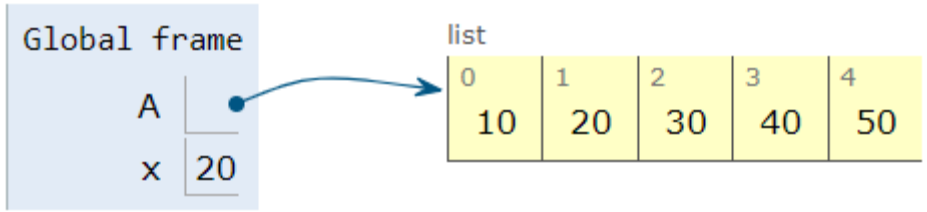
Reference slide

List indexing and slicing

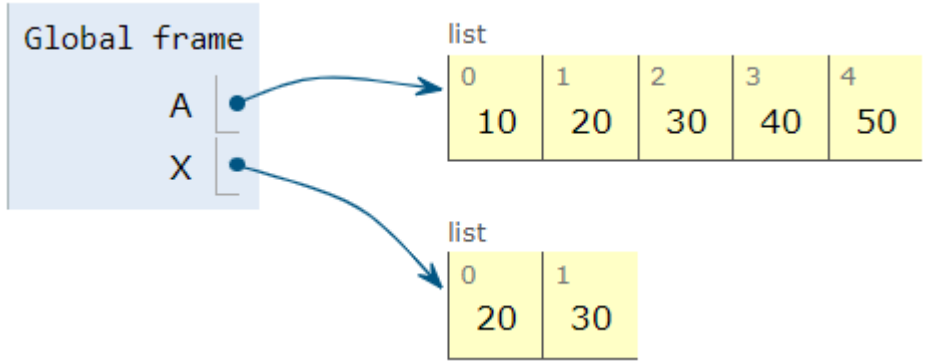
```
A = [10, 20, 30, 40, 50]
x = 99
A[1] = x
```



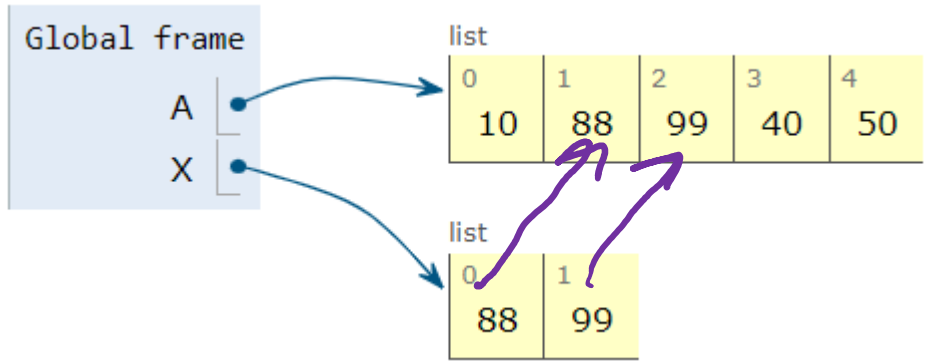

```
A = [10, 20, 30, 40, 50]
x = A[1]
```




```
A = [10, 20, 30, 40, 50]
X = A[1:3]
```



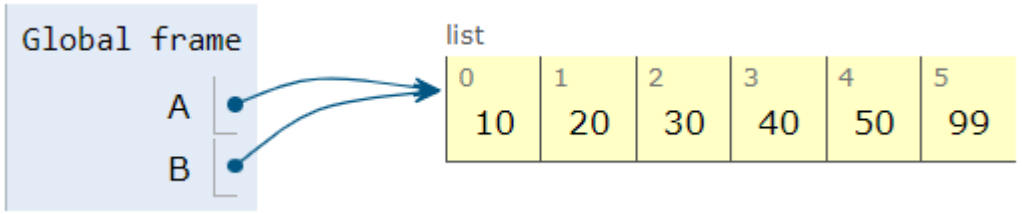

```
A = [10, 20, 30, 40, 50]
X = [88, 99]
A[1:3] = X
```



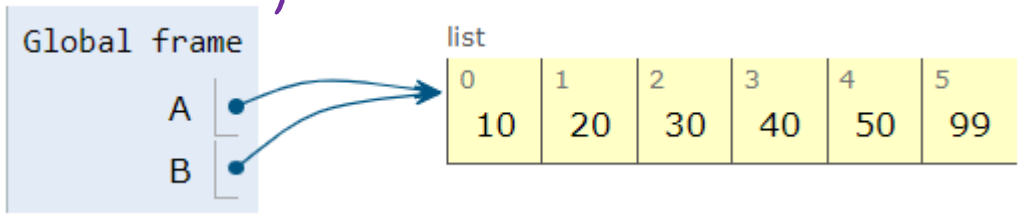
Adding elements

Reference slide

```
A = [10, 20, 30, 40, 50]
B = A
A.append(99)
```



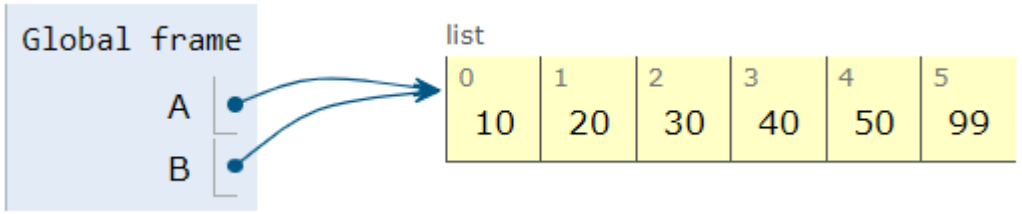

```
A = [10, 20, 30, 40, 50]
B = A
A += [99]
```



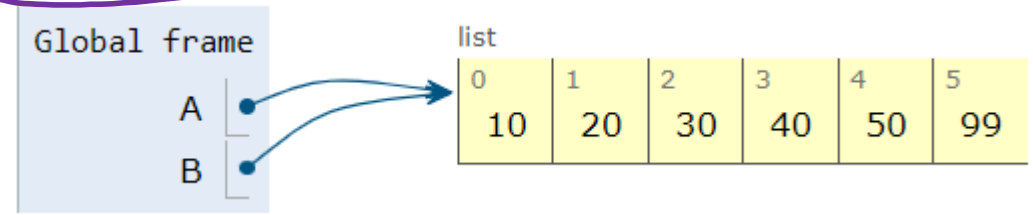
Adding elements

Reference slide

```
A = [10, 20, 30, 40, 50]
B = A
A.append(99)
```



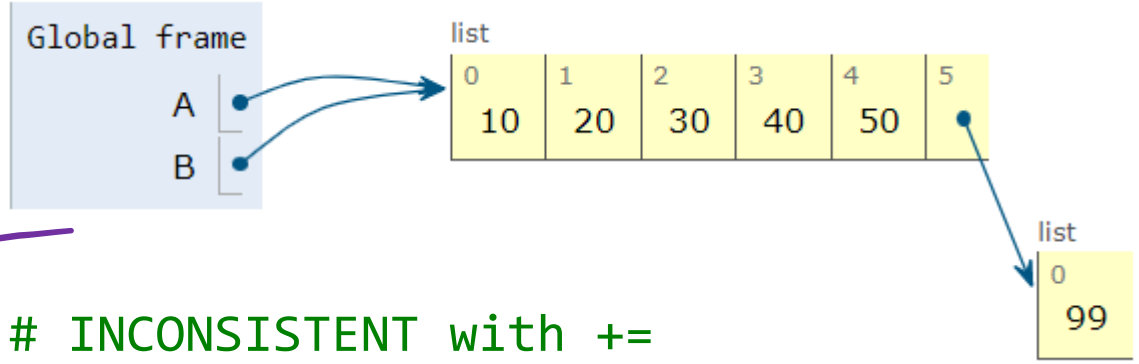

```
A = [10, 20, 30, 40, 50]
B = A
A += [99]
```



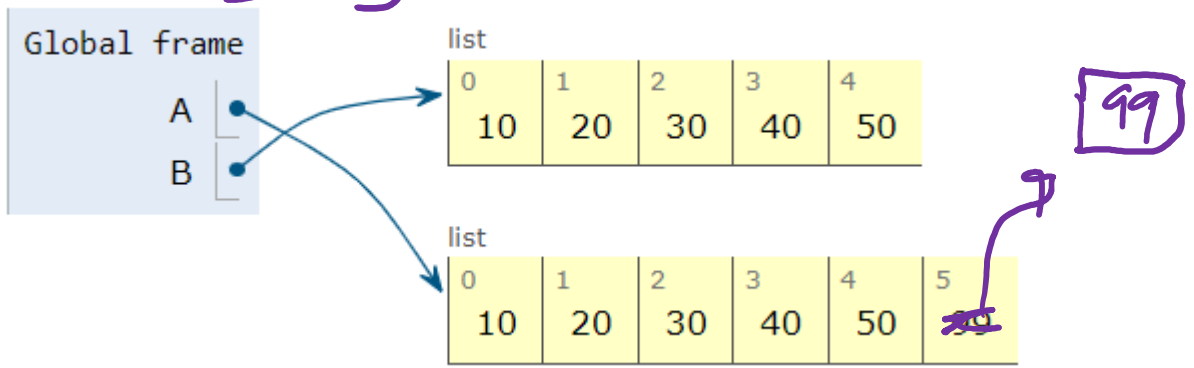

```
A = [10, 20, 30, 40, 50]
A += 99
TypeError: 'int' object is not iterable
```

```
# CAUTION
A = [10, 20, 30, 40, 50]
B = A
A.append([99])
```

[10, 20, 30, 40, 50, [99]]




```
# INCONSISTENT with +=
A = [10, 20, 30, 40, 50]
B = A
A = A + [99]
```



Poll 3

What are the resulting A, B, and C?

```
import copy

A = [10, 20, 30]
B = A
C = copy.copy(A)

A[0] = 44
B[1] = 55
C[2] = 66
A = A + [77]
```

I. A: [44, 20, 30, 77]
   B: [10, 55, 30]
   C: [10, 20, 66]

II. A: [44, 55, 30, 77]
   B: [44, 55, 30]
   C: [10, 20, 66]

III. A: [44, 20, 66, 77]
   B: [10, 55, 30]
   C: [44, 20, 66]

IV. A: [44, 55, 30, 77]
   B: [44, 55, 30, 77]
   C: [10, 20, 66]

Poll 4

What does this print?

```
def f(L):
    L.remove(3)

A = [2, 3, 4, 5]
print(f(A))
```

x = f(A)
print(x)

I. [2, 3, 4, 5]

II. [2, 4, 5]

III. [2, 3, 5]

IV. []

V. None

Poll 5

What does this print?

```
def f(L):
    L.remove(3)

A = [2, 3, 4, 5]
f(A)
print(A)
```

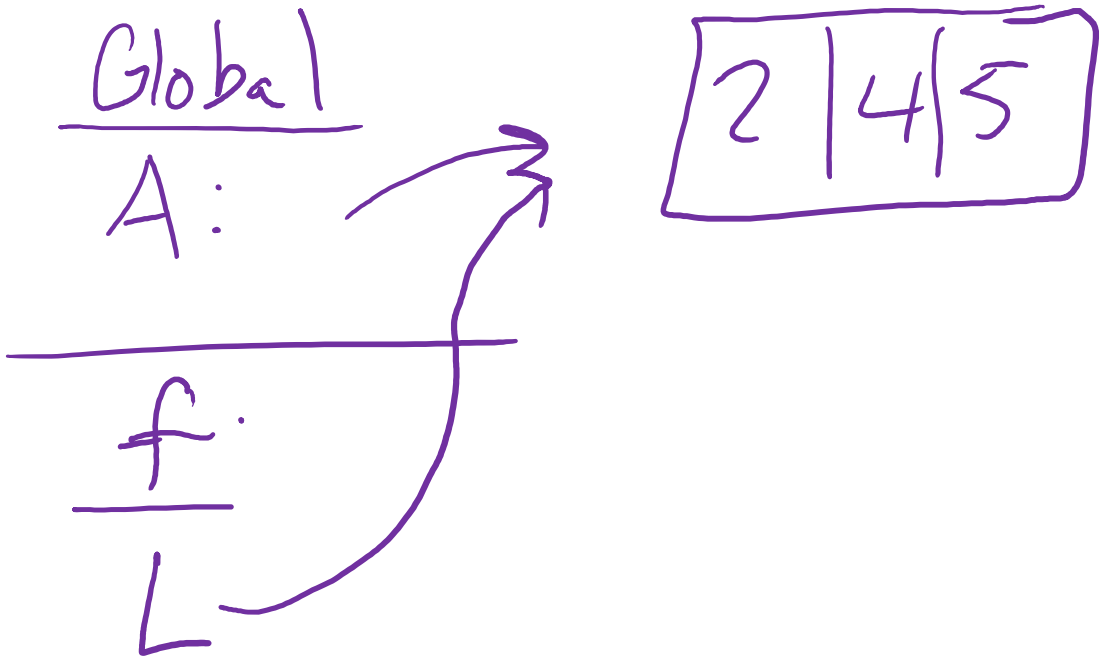
I. [2, 3, 4, 5]

II. [2, 4, 5]

III. [2, 3, 5]

IV. []

V. None



Poll 6

L = L*2 ← Nope Creates a second L and concatenates

Which is best?

I.

```
def doubleValues(L):
    for i in range(len(L)):
        L[i] *= 2
```

II.

```
def doubleValues(L):
    for i in range(len(L)):
        L[i] *= 2
    return L
```

Modifies L and returns L??

III.

```
def doubleValues(L):
    A = []
    for item in L:
        A.append(item*2)
    return A
```

IV.

```
def doubleValues(L):
    A = []
    for item in L:
        A.append(item*2)
```

Creates A but doesn't return

Poll 6

Which is best?

I.

```
def doubleValues(L):
    for i in range(len(L)):
        L[i] *= 2
```

Destructive, Modifies L in-place

III.

```
def doubleValues(L):
    A = []
    for item in L:
        A.append(item*2)
    return A
```

Non-destructive, Creates and returns A