1. What is **similar** about lists and strings in Python?

2. What is **different** about lists and strings in Python?

3. What is the output of the following program? If there are any errors, correct them.

```python
a = 'apple'
b = 'banana'
c = 'carrot'
x = [a, b, c]
print(x[b])
```

4. What is the output of the following program? If there are any errors, correct them.

```python
my_list = ['one', 'two', 'three', 'four']
print(my_list[:3])
```
5. What is the output of the following program? If there are any errors, correct them.

```python
alist = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
print(alist)
del alist[3]
print(alist)
```

6. What is the output of the following program? If there are any errors, correct them.

```python
list_1 = ['b']
list_2 = ['a']
list_3 = ['c']
list_4 = list_3 + list_1 + list_1
print(list_4*3)
```

7. What is the output of the following program? If there are any errors, correct them.

```python
x = '1'
y = '3'
z = [1, 2, 3]
print(z)
z[2] = y
z += [x]
z += [x]
print(z)
```
8. What is the output of the following program? If there are any errors, correct them.

```python
m = ['x', 'y', 'z']
n = m
o = m[:]
print('m = ', m, '\n n = ', n, '\no = ', o)
n[1] = 6
print('m = ', m, '\n n = ', n, '\no = ', o)
```

9. What is the output of the following program? If there are any errors, correct them.

```python
list1 = [13, 17, 23]
list2 = list1
list2 = list2 + 37
list1.append(41)
list2 += '59'
print('list1 = ', list1, '\nlist2 = ', list2)
```

10. What is the output of the following program? If there are any errors, correct them.

```python
def foo(data):
    for i in data:
        print( i, end=''
    data += i
    print()
    return data

print( foo("hi") )

print( foo(["h", "i"]))
```
11. Write a function `find_min` that takes a list of numbers and returns the smallest number. You cannot use `min()` or any list functions.

Some examples:

```python
>>> find_min([12, 9, 10])
9
>>> find_min([12, 9, 10, 9])
9
>>> find_min([-12, 9, 10, -12, -12])
-12
```

12. Write a function `index_of_min` that takes a list of numbers and returns the first index of the smallest number. You cannot use `min()` or any list functions.

Some examples:

```python
>>> index_of_min([12, 9, 10])
1
>>> index_of_min([12, 9, 10, -7])
3
>>> index_of_min([-12, 9, 10, -12, -12])
0
```
13. Write a function `indexes_of_min` that takes a list of numbers and returns a list containing the indexes of all occurrences of the smallest number. You cannot use `min()` or any list functions.

Some examples:

```python
>>> indexes_of_min([12, 9, 10])
[1]
>>> indexes_of_min([12, 9, 10, 9])
[1, 3]
>>> indexes_of_min([-12, 9, 10, -12, -12])
[0, 3, 4]
```

14. Revise your game winner program from HW 04 to use lists. Recall that the instructions were:

Write a program to keep track of the scores players earn during three rounds of a game and compute the winner at the end. Your program will first ask the user for the number of players (out of a maximum of four). Then, for each round, the program will ask for each player’s score. The object of the game is to score 20 or more points, once one player scores that many points no more rounds are played. However, every player is allowed to complete the round before final scores are tallied. After the last round played, the final scores and the winner will be announced. Each player’s final score is the sum of the scores that player earned over all played rounds of the game, and the winner is the player with the largest final score.

Note: Your program should use a single `for` loop.