1) How are lists like strings?:

   a. What is the result of the following code:

      ```python
      a = ["soccer", 6, 8.32]
      print(len(a))
      ```

   b. What is the result of the following code:

      ```python
      b = ["double", "triple", [8, 10, 12], [], 4.5]
      print(len(b))
      ```

   c. What is the result of the following code?

      ```python
      print(b[1])
      ```

   d. What is the result of the following code?

      ```python
      print(b[2][1])
      ```
e. What is the result of the following code?

```python
a = ["soccer", 6, 8.32]
print(6 in a)
```

f. What is the result of the following code?

```python
print( 10 in b )
```

g. What is the result of the following code?

```python
c = [22, 31, 43]
d = [18, 19, 20]
print(c + d)
```

h. What is the result of the following code?

```python
print(d * 2 + c)
```
i. What is the result of the following code?

```python
e = ["poppy", "rose", "petunia", "daisy"]
f = [8, 10, [‘a’, ‘b’, ‘c’], “twenty”, []]
g = e[1:2] * 2 + f[2:]
print(g)
```

2) How are lists not like strings?

a. What does the following code do?

```python
names = [“charlie”, “twenty”, “keni”, “lara”]
print(names[2])
names[1] = “meg”
print(names)
```

b. What does the following code do?

```python
list_a = [4, 5, 6, 7, 8]
list_b = [3, 4, 5]
list_a[2:2] = list_b
print(list_a)
```
c. What does the following code do?

```python
list_a = [4, 5, 6, 7, 8]
del list_a[2:4]
print(list_a)
```

d. What does the following code do?

```python
list_a = [6, 7, 8]
list_b = [6, 7, 8]
print(list_a is list_b)
print(list_a == list_b)
```
e. What happens in the following code?

```python
def print_code():
    list_a = [6, 7, 8]
    list_b = [6, 7, 8]
    print(list_a is list_b)
    print(list_a == list_b)

    list_a = list_b
    print(list_a is list_b)
    print(list_a == list_b)

    list_b[1] = 200
    print(list_a)
```

f. What can we do if we want `list_b` to be a copy of `list_a` but be a separate list so that when we modify `list_b`, `list_a` does not change?
3) Using Lists

a. Write a function that takes a list and returns a new list with only the items that were at an odd index of the original.

b. 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.