Chapter 10 Lists

1. emptyList = []
   lst = [1, 32, 2]

2.

There are 6 elements in the list. The index of the first element is 0. The index of the last element is 5. lst[2] is 12. list[-2] is list[6-2], which is 10.

3.

```
lst.append(40) => [30, 1, 2, 1, 0, 40]
lst.insert(1, 43) => [30, 43, 1, 2, 1, 0]
lst.extend([1, 43]) => [30, 1, 2, 1, 0, 1, 43]
lst.remove(1) => [30, 2, 1, 0]
lst.pop(1) => [30, 2, 1, 0]
lst.pop() => [30, 1, 2, 1]
lst.sort() => [0, 1, 1, 2, 30]
lst.reverse() => [0, 1, 2, 1, 30]
random.shuffle(list) => list is randomly shuffled
```

4.

```
lst.index(1) => 1
lst.count(1) => 2
len(list) => 5
max(list) => 30
min(list) => 0
sum(list) => 34
```

5.

```
list1 + list2  => [30, 1, 2, 1, 0, 1, 21, 13]
2 * list2 => [1, 21, 13, 1, 21, 13]
list2 * 2 => [1, 21, 13, 1, 21, 13]
list1[1 : 3] => [1, 2]
list1[3] => 1
```

6.

```
[x for x in list1 if x > 1]  => [30, 2]
[x for x in range(0, 10, 2)]  =>  [0, 2, 4, 6, 8]
[x for x in range(10, 0, -2)] => [10, 8, 6, 4, 2]
```

7.

```
list1 < list2  =>  False
list1 <= list2 =>  False
list1 == list2 =>  False
list1 != list2 =>  True
list1 > list2 =>  True
list1 >= list2 =>  True
```

8.

Indicate true or false for the following statements:

- Every element in a list must have the same type. (False)
- The list size is fixed after it is created. (False)
- The list can have duplicated elements. (True)
- The elements in a list can be accessed via an indexer. (True)

9.

list1 is [22, 43] and list2 is [22, 43]

10.

```
list1 is [22, 43]
list2 is [1, 43]
```

11

```
list(s)

s1.split('o')  is ['Welc', 'me']
```

12.

a. `lst = 100 * [False]`
b. `lst.append(5.5)`
c. `print(lst[0] + lst[1])`
d. `total = sum(lst[0:5])`
e. `minimum = min(lst)`
f. `print(list[random.randint(0, len(lst) - 1)])`

13.

Causes a run time error (index out of range)

14.

```
[1, 1, 1, 1, 1, 1]
```

15.

```
[111, 3, 5, 7, 9]
[111, 3, 5, 7, 9]
```

16.

```
[111, 3, 5, 7, 9]
[1, 3, 5, 7, 9]
```

17.

No. The reference of the list is passed to the parameter in the function. No new list is created.

18.

(a)

```
number is 0 and numbers[0] is 3
```

(b)

```
1 2 3 4 5
```

This is a tricky question. You need to read the code carefully to get the correct output. Note that the reverse function is supposed to reverse the list, but it was coded incorrectly so it did not reverse the original list. In order to reverse the elements in the original list, modify the function as follows:

```
def reverse(lst):
    for i in range(int(len(lst) / 2)):
        temp = lst[i]
        lst[i] = lst[len(lst) - 1 - i]
        lst[len(lst) - 1 - i] = temp
```

19.

(a)

```
[1, 2, 3]
[2, 3]
```

(b)

```
[1, 2, 3]
[1, 2, 3]
```

20. Omitted

21 (a) No. (b) 3

22 Omitted

23 Omitted

24 Simply change (currentMax < lst[j]) in line 9 to (currentMax > lst[j])

25 Simply change lst[k] > currentElement in line 8 to lst[k] < currentElement