Quiz 06

Due Mar 4 at 10pm **Points** 10 **Questions** 5 **Time Limit** None

Instructions

Answer the following questions in your own words. Do NOT simply cut and paste the information from the slides. You will receive a score of 0 if you copy the prose from the slides.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	60 minutes	10 out of 10

Correct answers are hidden.

Score for this quiz: **10** out of 10
Submitted Mar 4 at 9:49pm
This attempt took 60 minutes.

Question 1

2 / 2 pts

Describe the four types of Python containers. How are they different from each other?

Your Answer:

Python Containers :

1. Lists: Lists have arbitrary values with mutable elements and the elements are ordered.

2. Tuples: Tuples are also ordered like lists but are the elements of the tuple are immutable that means the elements once defined can not be changed.

3. Dictionaries: Dictionaries have a unique key for every value, it contains a pair of key, value. Also, the dictionaries are mutable, unlike tuples.

4. Sets: Sets are unordered and have all unique elements in them and are also mutable.

Lists are ordered and mutable with elements of any type.

Tuples are ordered, but not mutable with elements of any type.

Dicts map hashable keys to values.   Dicts are mutable.

Sets contain distinct hashable values and are mutable.

Question 2

2 / 2 pts

Explain the difference between list.append() and list.extend(). Include an example in your answer.

Your Answer:

1. list.append(): list.append() adds new values at the end of the list.

2. list.extend(): list.extend() concatenates two lists.

For example:

lst=['a', 'b', 'c']

lst.append(['1', '2'])

Output: ['a', 'b', 'c', ['1', '2']]

lst=['a', 'b', 'c']

lst.extend(['d'])

Output: ['a', 'b', 'c', 'd']

list.append(value) appends the value a new element at the end of the list.

list.extend(sequence) concatenates list and the sequence.

Question 3

2 / 2 pts

Write a code fragment that shows how to emulate the behavior of list.remove(value) using only list.pop() and list.index()

Your Answer:

```
def lst_remove(i,lst):
          if i in lst:
                a= lst.index(i)
                 return(lst.pop(a))


lst_remove(1,[1,3,2,4])
```

```
def my_remove(t, l):
    if t in l:
       l.pop(l.index(t))
```

Question 4

2 / 2 pts

What is the difference between sorted(list) and list.sort()? Show examples of both.

Your Answer:

sorted(list) doesn't make any changes in the original list and returns a copy of the list in a new list with the sorted elements.

list.sort() modifies the original list and sorts it.

lst=[6,4,5]

sorted(lst)

Output: [4,5,6]

lst.sort()

Output: lst=[4,5,6]

sorted(list) returns a sorted **copy** of the list

list.sort() sorts the list in place

Example from slides:

```
lst1 = [3, 1, 2]
sorted(lst1) returns a new list [1, 2, 3].  lst1 is not changed

lst1.sorted() sorts lst1 in place, changing the list so
lst1 == [1, 2, 3]
```

Question 5

2 / 2 pts

Consider the code fragment:

```
x = [1, 2, 3]
y = [x, x]
x[0] = 4
```

What is the value of y? Why?

Your Answer:

y=[[4,2,3],[4,2,3]]

Because x[0] = 4 changes the value of x to [4,2,3] from [1,2,3] and y=[x,x] thus y contains x which changes the value of y too

```
[[4, 2, 3], [4, 2, 3]]
```

Y consists of two instances of x so changing x also changes y.

Quiz Score: **10** out of 10