Chapter 8

Question 1

How are "collection" variables different from normal variables?

○ Collection variables pull multiple network documents together

○ Collection variables merge streams of output into a single stream

○ Collection variables can only store a single value

◉ Collection variables can store multiple values in a single variable

---

Question 2

What are the Python keywords used to construct a loop to iterate through a list?

○ try / except

◉ for / in

○ foreach / in

○ def / return

Question 3

For the following list, how would you print out 'Sally'?

```
friends = [ 'Joseph', 'Glenn', 'Sally' ]
```

○ print friends[3]

◉ print(friends[2])

○ print(friends[2:1])

○ print(friends['Sally'])

Question 4

What would the following Python code print out?

```
fruit = 'Banana'
fruit[0] = 'b'
print(fruit)
```

○ banana

○ B

● Nothing would print - the program fails with a traceback error

○ Banana

○ b

○ [0]

Question 5

Which of the following Python statements would print out the length of a list stored in the variable **data** ?

○ print(strlen(data))

○ print(data.length())

○ print(data.length)

○ print(length(data))

● print(len(data))

○ print(data.Len)

Question 6

What type of data is produced when you call the **range()** function?

```python
x = list(range(5))
```

○ A list of characters

○ A boolean (true/false) value

○ A list of words

● A list of integers

○ A string

Question 7

What does the following Python code print out?

```python
a = [1, 2, 3]
b = [4, 5, 6]
c = a + b
print(len(c))
```

○ [1, 2, 3]

○ [4, 5, 6]

○ 21

○ 15

● 6

○ [1, 2, 3, 4, 5, 6]

Question 8

Which of the following slicing operations will produce the list [12, 3]?

```
t = [9, 41, 12, 3, 74, 15]
```

- ○ t[1:3]
- ◉ t[2:4]
- ○ t[2:2]
- ○ t[12:3]
- ○ t[:]

Question 9

What list method adds a new item to the end of an existing list?

- ○ index()
- ○ add()
- ○ pop()
- ○ push()
- ○ forward()
- ◉ append()

Question 10

What will the following Python code print out?

```
friends = [ 'Joseph', 'Glenn', 'Sally' ]
friends.sort()
print(friends[0])
```

- ◉ Glenn
- ○ friends
- ○ Sally
- ○ Joseph