Chapter 8

Total points 10

Question 1

How are "collection" variables different from normal variables? **1 / 1 point**

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 **1 / 1 point**

def / return

foreach / in

for / in

Question 3

For the following list, how would you print out 'Sally'? **1 / 1 point**

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

print(friends['Sally'])

print(friends[2])

print friends[3]

print(friends[2:1])

Question 4

What would the following Python code print out? **1 / 1 point**

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

Nothing would print - the program fails with a traceback error

B

banana

Banana

[0]

b

Question 5

Which of the following Python statements would print out the length of a list stored in the variable *data*? **1 / 1 point**

print(length(data))

print(strlen(data))

print(data.length)

print(len(data))

print(data.length())

print(data.Len)

Question 6

What type of data is produced when you call the *range()* function? **1 / 1 point**

A list of characters

A list of integers

A boolean (true/false) value

A string

A list of words

Question 7

What does the following Python code print out? **1 / 1 point**

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

[1, 2, 3]

15

[1, 2, 3, 4, 5, 6]

[4, 5, 6]

6

21

Question 8

Which of the following slicing operations will produce the list [12, 3]? **1 / 1 point**

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

t[:]

t[1:3]

t[2:4]

t[2:2]

t[12:3]

Question 9

What list method adds a new item to the end of an existing list? **1 / 1 point**

add()

pop()

append()

index()

push()

forward()

Question 10

What will the following Python code print out? **1 / 1 point**

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

friends

Glenn

Sally

Joseph