

1. How are "collection" variables different from normal variables?

1 point

- ☐ Collection variables merge streams of output into a single stream
- ☒ Collection variables can store multiple values in a single variable
- ☐ Collection variables can only store a single value
- ☐ Collection variables pull multiple network documents together

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

1 point

- ☐ def / return
- ☒ for / in
- ☐ try / except
- ☐ foreach / in

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

1 point

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

- ☐ print(friends[2:1])
- ☐ print(friends['Sally'])
- ☒ print(friends[2])
- ☐ print friends[3]

4. What would the following Python code print out?

1 point

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

- ☐ B
- ☐ banana
- ☒ Nothing would print - the program fails with a traceback error
- ☐ b
- ☐ Banana
- ☐ [0]

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

1 point

- ☒ `print(len(data))`
- ☐ `print(data.Len)`
- ☐ `print(length(data))`
- ☐ `print(data.length)`
- ☐ `print(strlen(data))`
- ☐ `print(data.length())`

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

1 point

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

- ☐ A list of words
- ☒ A list of integers
- ☐ A list of characters
- ☐ A string
- ☐ A boolean (true/false) value

7. What does the following Python code print out?

1 point

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

- ☐ 15
- ☒ 6
- ☐ [1, 2, 3]
- ☐ [1, 2, 3, 4, 5, 6]
- ☐ [4, 5, 6]
- ☐ 21

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

1 point

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

- ☐ t[1:3]
- ☐ t[:]
- ☐ t[2:2]
- ☐ t[12:3]
- ☒ t[2:4]

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

1 point

- ☐ add()
- ☐ push()
- ☐ forward()
- ☒ append()
- ☐ index()
- ☐ pop()

10. What will the following Python code print out?

1 point

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

- ☐ Sally
- ☐ friends
- ☐ Joseph
- ☒ Glenn