

Name: _____ Partners: _____

Python Activity 18: More about Lists and Strings

Critical Thinking Questions:

Learning Objectives
 Students will be able to:

Content:

- Explain how to send a list as an argument to a function
- Explain the purpose of these functions: lower(), upper(), strip(), replace(), sorted(), join()
- Demonstrate the use of **slicing** with strings and lists.

Process:

- Write code that uses the following functions: lower(), upper(), strip(), replace(), sorted(), join()
- Write code that uses **slicing** to access elements of strings and lists

Prior Knowledge

- Python concepts from Activities 1-16

1. Examine the following program and its output. It includes a function that takes a list as an argument.

<pre> 2 def orderList(anyList): 3 newList = sorted(anyList) 4 newList = newList[::-1] 5 return newList 6 7 myList = [] 8 for y in range(10): 9 n = int(input("Gimme an int: ")) 10 myList.append(n) 11 print(orderList(myList)) </pre>	<pre> Gimme an int: 6 Gimme an int: 2 Gimme an int: 99 Gimme an int: 1 Gimme an int: 7777 Gimme an int: -34 Gimme an int: 0000 Gimme an int: 5 Gimme an int: 7 Gimme an int: 2 [7777, 99, 7, 6, 5, 2, 2, 1, 0, -34] </pre>
--	--

- a. What is the name of the function defined in this program? _____
- b. What does the function do? _____
- c. What might `newList = sorted(anyList)` do? _____
- d. What might `newList[::-1]` do? _____

2. Examine the following code:

```

14 usrNoun = input("Gimme a plural noun: ")
15 madlib = "The mountains! The mountains! We greet them with a song!"
16 mSentence = madlib.replace('mountains', usrNoun)
17 print(mSentence)
                
```

- a. What inputs might you enter to see what the program does?

- b. Examine the output for some sample inputs below.

```
Gimme a plural noun: students
The students! The students! We greet them with a song!
Gimme a plural noun: CATS
The CATS! The CATS! We greet them with a song!
Gimme a plural noun: toDay200224
The toDay200224! The toDay200224! We greet them with a song!
```

What does the program do?

- c. What does the **replace()** function do? How do the first & second parameter differ?

3. Examine the following code.

```
20 befString = input("Enter a string with some spaces: ")
21 aftString = befString.strip()
22 print(aftString, len(befString), 'vs', len(aftString))
```

- a. What are some inputs you might use to see what the program does?

- b. Examine the output from the program below.

```
Enter a string with some spaces: hello world
hello world 11 vs 11
Enter a string with some spaces:      hello world
hello world 15 vs 11
Enter a string with some spaces:      hello world
hello world 12 vs 11
Enter a string with some spaces: hello world
hello world 30 vs 11
```

What does the program do?

- c. What does the **strip()** function do?

4. Examine the following code.

```
30 sentence = "This is a sentence with some spaces."
31 numSpaces = 0
32 for index in range(len(sentence)):
33     if sentence[index].isspace():
34         numSpaces += 1
35 print("There are", numSpaces, "spaces in the sentence.")
```

- a. What does the program do? _____

- b. What does the **isspace()** function do? _____
- c. If we replace the call to **isspace()** with **isalpha()** the program counts '29'. What might **isalpha()** do? _____
- d. If we replace the call to **isalpha()** with **isdigit()** the program counts '0'. What might **isdigit()** do? _____
- e. How could we rewrite line 32 to not use **range()**? What other line would we have to change? _____

5. Examine the following code and its output:

```
38 username = input("Enter user name: ")
39 if username.upper() == "CSCI134":
40     print("Correct!")
41 else:
42     print("Invalid user name.")
```

```
Enter user name: Csci134
Correct!
Enter user name: csci134
Correct!
Enter user name: CSCI376
Invalid user name.
Enter user name: CSCI134
Correct!
```

- a. For each of the following inputs, what might the result of line 39 be?
- Csci134 _____
 - csci134 _____
 - CSCI376 _____
 - CSCI134 _____

- b. What does the **upper()** function do? _____
- c. Use the **lower()** function instead of the **upper()** function in the program above. Revise the line of code so that it still produces the *same* output. Execute the program again with the data listed in 'a'. (Write the revised code below.)

6. Examine the following code and its output:

```
40 emadd = input("Email address? ")
41 print("Split:", emadd.split('.'))
42 nospam = ' DOT '.join(emadd.split('.'))
43 print("Spam free:", nospam)
```

```
Email address? iris@cs.williams.edu
Split: ['iris@cs', 'williams', 'edu']
Spam free: iris@cs DOT williams DOT edu
```

- a. What does **split()** do? _____
- b. What does the argument passed to **split()** represent? _____
- c. What does **join()** do? _____
- d. What does the argument passed to **join()** represent? _____
- e. What does the object right before **.join()** represent? _____

FYI: **Slicing** is a technique available in Python that allows you to access parts of lists or strings. You can select multiple elements of a list or string.
Syntax: <listOrStringName>[indexOfFirstItem : indexAfterLastItem].

7. In this section we are going to try to access parts of a **string** using **slicing**. Enter and execute the following code. Examine the syntax of the code. It uses slicing to access parts of a string.

```
courseName = 'Introduction to Computer Science'  
print(courseName)  
print(courseName[0])  
print(courseName[-2])  
print(courseName[0:13])  
print(courseName[16:24])  
print(courseName[25:])
```

a. What is the output for each print statement in the program?

b. The first three **print** statements should be familiar. What does the fourth print statement do? Explain the meaning of **[0:13]**. `print(courseName[0:13])`

c. What does the following print statement do? Explain the meaning of **[16:27]**.

`print(courseName[16:27])`

d. What does the following print statement do? Explain the meaning of **[28:]**.

`print(courseName[28:])`

8. Finally, examine **slicing** using **lists**. Enter and execute the following program.

```
courselist = ['CSCI134', 'CSCI136', 'CSCI237', 'CSCI256']  
print(courselist)  
copylist = courselist[:]  
print(copylist)  
copylist[1] = "CSCI334"  
print(copylist)  
print(copylist[1:3])
```

a. What is the output of each print statement in the program?

b. Explain what the following code does. `copyList = courseList[:]`

c. Explain what the following code does: `copylist[1] = "CSCI334"`

d. Explain what the following code does: `print(copyList[1:3])`

Application Questions: Use the Python Interpreter to check your work

1. Create a list named "Days" that includes all the days of the week. Print the list.

2. Create a line of code that uses **slicing** to print the last three days in the list "Days" which you created in question 1.

3. Create a list named "Vowels" that includes the vowels 'a', 'e', 'i', 'o', 'u'.

4. Use the code in question 3, and create a program that analyzes a user's input. Complete the following steps:

a. Create code that prompts the user for a vowel.

b. Create code to determine if the user input is a vowel. If so, congratulate them.

c. Create code that determines if it is a letter, but not a vowel and prints a message that indicates that.

d. Add code that determines if the user input is a digit instead of a letter, print a message that indicates that as well.

e. Otherwise, tell the user that their input was not a vowel, a letter, or a number.

f. Prevent the program from crashing by terminating the program if the user enters more than one character. This should actually be tested first.

g. Put all the code together and test the program with several sets of data. List a sample output.
