Lab Exercise 9

Focus

1. Strings and string manipulation

This lab maps to learning objectives 1.1 through 1.6 in Competency Module Nine – Competency Nine - Write a Working Program that Manipulates Strings including Slicing, Searching, Splitting, and Case Conversions

Part 0: A Note on Validation Loops

The examples of validation loops in the textbook all use a priming read like this example:

```
grade = int(input(‘Enter the grade: ‘))
while grade < 0 or grade > 100:
    print(‘error – grade must be 0 through 100’)
    grade = int(input(‘Enter the grade: ‘))
#code to process the valid grade goes here
```

In situations where the validation condition is more complex or where you want to have more than one error message, you might want to use a Boolean variable (also called a flag) in the validation loop. For example:

```
valid = False
while not valid:
    grade = int(input(‘Enter the grade: ‘))
    if grade < 0:
        print(‘error – grade must be zero or larger’)
    elif grade > 100:
        print(‘error m- grade must be 100 or less’)
    else:
        valid = True;
#code to process the valid grade goes here
```

Part A: Write Something New

Write a complete and syntactically correct Python program to solve the following problem:

Student email addresses at Austin Community College are in the format
firstname.lastname@g.austincc.edu
Your program should input and validate an email address. Then it will extract the student’s name and print it. Here are more detailed instructions:

1. Input the email address in a validation loop.. **(LO 1.1, 1.5)**
2. To validate the address, you will need to search for the position of the first period and the @ sign. The address is valid if it contains both a period and an @ sign, and all alphabetic characters are lower case. If an invalid address is entered, print appropriate error message(s) and have the user input another address. **(LO 1.6)**
3. Once a valid address has been obtained, extract the student's first name and last name from the email address.
4. Display the student's name to the monitor. Be sure to capitalize the first letter of the first and last names. **(LO 1.3, 1.5)**

Save your file as *firstname_lastname_Lab9a.py* where you will replace firstname and lastname with your actual first name and last name. Remember to use the extension .py.

Run and test your program for all conditions. Once you are sure it works you will turn in the items listed in the last section.

Part B: Write Something Else New

Write a complete and syntactically correct Python program to solve the following problem:

1. In a validation loop, input a date in numeric format from the user (e.g. mm/dd/yy). **(LO 1.1)**
2. The date is valid if all the following are true. The month (mm) must be an integer in the range 1 through 12. The day (dd) must be an integer in the range 1 through 31. The year (yy) must be ttwo digits and it must be this year (16). If the date is invalid, display an error message and ask for input again. **(LO 1.2, 1.4)**
3. Once the program has obtained a valid date, output the date in long date format. Thus a string that was input as 06/01/16 will be output as June 1, 2016. **(LO 1.3, 1.5)**

Please save your file as firstname_lastname_Lab9b.py where you will replace firstname and lastname with your actual first name and last name. Remember to use the extension .py.

Run and test your program for all conditions. Once you are sure it works you will turn in the items listed in the next section.

Turn In

All labs will be graded in Blackboard. Once you are done with the lab turn it in to the Lab 9 link. Please read the How To Submit instructions if you have any questions or contact the instructor. For this lab you will turn into Blackboard:

1. The Python *code file* you saved in part A.

2. The Python code file you saved in part B.