Assignment 7

Name: Lukas Pfalz

ITS 1110

Section: CSC-1110-71762

# **Assignment 7**

***COMPLETE THE HONOR CODE BELOW***

**HONOR CODE:**

I pledge that this program represents my own program code, I have received help from no one and I have given help to no one.

**OR**

I received help from **NAME OR NO ONE** in designing and debugging my program.

I have given help to **NAME OR NO ONE** in designing and debugging my program.



This assignment is required.

The grading form shows point values for this assignment. Please review it now.

Show screen shots of the python code with comments and your input/output window.

You should use several well-planned sets of data to check out your program. Testing your programs with just the data that is asked for in the assignment does not necessarily mean that the programs will work for all cases.

Please include the following comments in each of your Python programs:

Your Name

Section

Date

Description

Assignment Number

A python template (python\_template.py) has been provided for you to use.

Name this document XXX\_Assignment 7 where XXX are your initials. Include a python file named P01.py, P02.py, etc. for each problem.

You have been hired by a small software company to create a "thesaurus" program that replaces words with their synonyms. The company has set you up with a sample thesaurus stored in a Python dictionary object. Here's the code that represents the thesaurus:

# define our simple thesaurus

thesaurus = {

"happy": "glad",

"sad" : "bleak"

}

The dictionary contains two keys - "happy" and "sad". Each of these keys holds a single synonym for that key.

## Write a program that asks the user for a phrase. Then compare the words in that phrase to the keys in the thesaurus. If the key can be found you should replace the original word with a random synonym for that word. Words that are changed in this way should be printed in UPPERCASE letters. Make sure to remove all punctuation from your initial phrase so that you can find all possible matches.

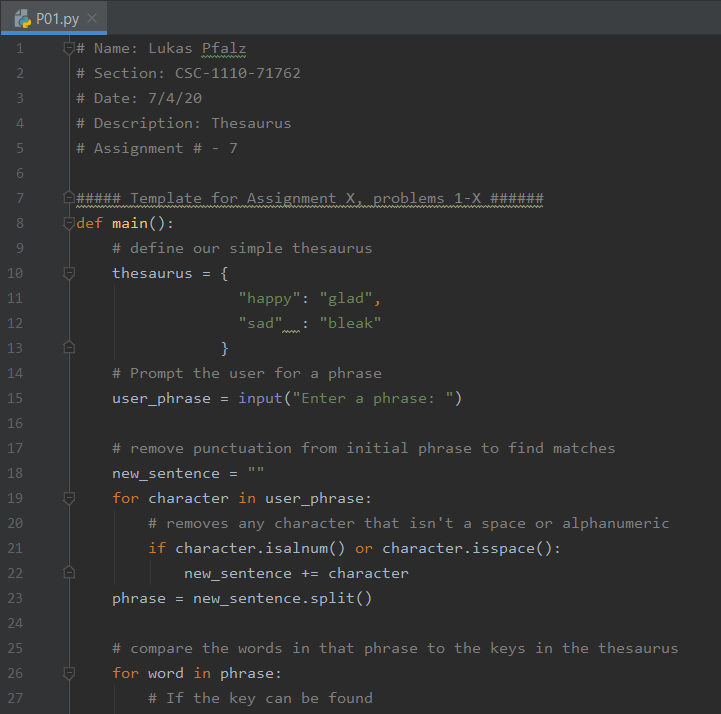
Here's the sample run of your program:

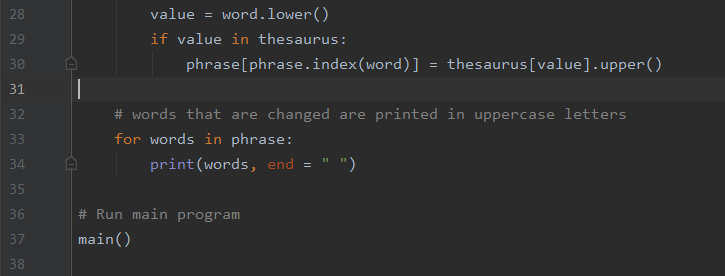
Enter a phrase: Happy Birthday! exclaimed the sad, sad kitten

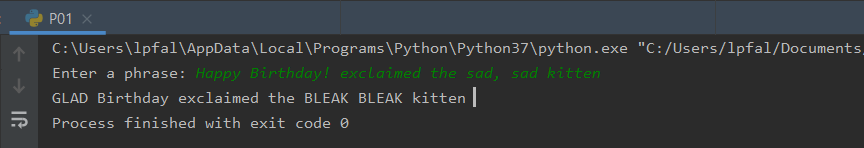
GLAD birthday exclaimed the BLEAK BLEAK kitten

Comment your source code and describe your code to someone who may be viewing it for the first time.

**PLACE SCREEN SHOTS OF THE PYTHON CODE AND ALL I/O BELOW.**







## Numerology is the "study of the purported mystical or special relationship between a number and observed or perceived events." It has been used throughout human history as a way to attach meaning to a name, object or event using mathematics. It is considered a "pseudoscience" by modern scientists since it has no basis in observable phenomena. With that said, it makes a great programming challenge so we're going to go with it! :)

## What you want to do for this project is to ask the user to type in their name. Next, you will need to use a technique called "theosophical reduction" to convert their name into a number. With this technique we assign each letter of the alphabet its own number. For example, the letter "a" is equal to the number 1. "b" = 2, "c" = 3, "z" = 26, etc. You should ignore non-alphabetic characters (i.e. numbers, spaces and special characters)

## Once you've gotten all of the letters converted into numbers you can add them up into one single number. This is the "numerology number" for the name that the user entered.

So, for the name "craig" the numerology number would be:

c = 3

r = 18

a = 1

i = 9

g = 7

3 + 18 + 1 + 9 + 7 = 38

Here's the sample input and output your program should produce.

Name: craig

Your 'cleaned up' name is: craig

Reduction: 38

Name: craig kapp

Your 'cleaned up' name is: craigkapp

Reduction: 82

Name: rumple stil skin

Your 'cleaned up' name is: rumplestilskin

Reduction: 198

Name: !rumple!stil!skin

Your 'cleaned up' name is: rumplestilskin

Reduction: 198

Name: pikachu!pikapika!

Your 'cleaned up' name is: pikachupikapika

Reduction: 143

Name: PIKACHUpikapika

Your 'cleaned up' name is: pikachupikapika

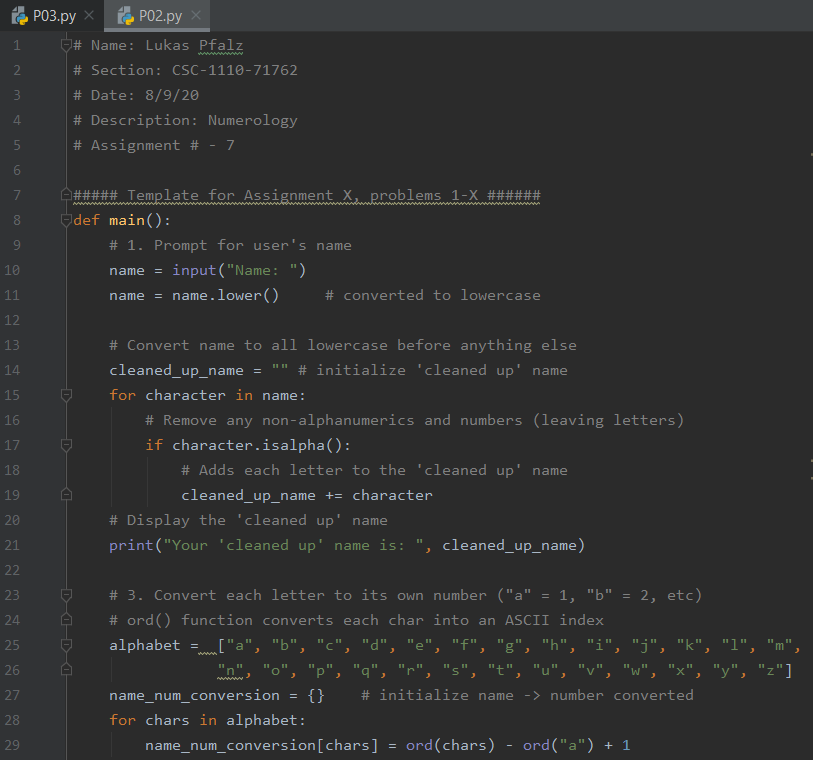
Reduction: 143

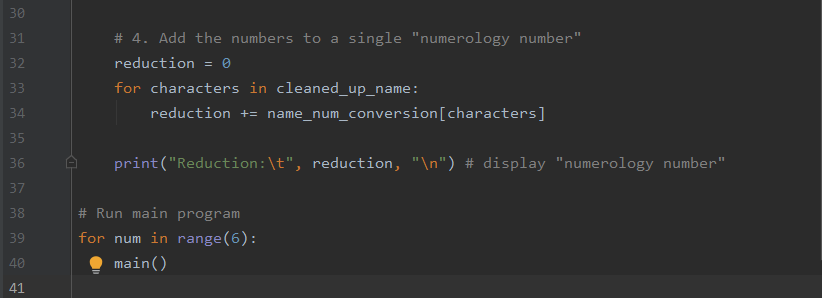
Some hints:

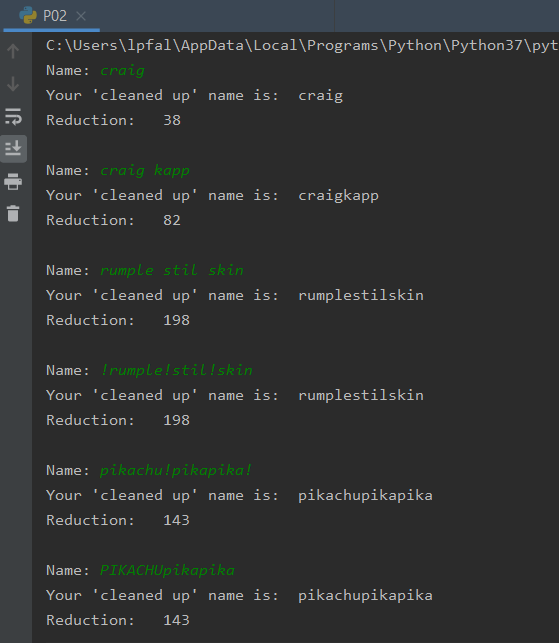
* Convert the user's name to all uppercase or all lowercase before you do anything else
* Remove any spaces, numbers or special characters from the name to ensure that you are only working with the letters A-Z
* The ord() function may be userful to convert each character into an ASCII index

Comment your source code and describe your code to someone who may be viewing it for the first time.

**PLACE SCREEN SHOTS OF THE PYTHON CODE AND ALL I/O BELOW.**







## Write a class named Employee that holds the following data about an employee in attributes: name, ID number, department, and job title. The class should have Accessors and Mutators for each field. Then write a program that creates three Employee objects to hold the following data:

| **Name** | **ID Number** | **Department** | **Job Title** |
| --- | --- | --- | --- |
| Susan Meyers | 47899 | Accounting | Vice President |
| Mark Jones | 39119 | IT | Programmer |
| Joy Rogers | 81774 | Manufacturing | Engineer |

Comment your source code and describe your code to someone who may be viewing it for the first time.

**PLACE SCREEN SHOTS OF THE PYTHON CODE AND ALL I/O BELOW.**

