**4CS001**

**Python Workshop 7: Dictionaries and Sets**

These are instructions for your seventh Python workshop. These workshops are designed for you to be able to make a good start on in your weekly lab sessions, but you may need to take them home to finish. There are some tasks that will not have been covered in the lecture, these will require you to do some independent research!

When you have finished this workshop, please upload your work to Canvas. This way you can always check back later to see how you solved a problem.

**Select:** Start > All Programs > IDLE (Python GUI)

**Part 1**

**1.** Create three dictionaries:

dic1 = {1:10, 2:20}  
dic2 = {3:30, 4:40}  
dic3 = {5:50, 6:60}

**(a)** Write code to concatenate these dictionaries to create a new one. Create a variable called nums to store the resulting dictionary. There are multiple ways to do this, however, one of the easiest is to convert each of the dictionaries items to a list (which can be added together) and pass them to the dict() constructor.

**(b)** Write code to add a new key/value pair to the dictionary nums: (7, 70)

**(c)** Write code to update the value of the item with key 3 in nums to 80

**(d)** Write code to remove the third item from dictionary nums.

**(e)** Write code to sum all the items in the dictionary nums

**(f)** Write code to multiply all the items in the dictionary nums

**(g)** Write code to retrieve the maximum and minimum values in nums.

**2.** Create two sets:

set1 = {20, 40, 60}  
set2 = {10, 20, 30, 40, 50, 60}

**(a)** Write code to perform a union of these sets. Print the length of the resulting set.

**(b)** Write code to perform an intersection of set1 and set2.

**(c)** Write code to compute the symmetric difference between set1 and set2

**(d)** Write code to add the value 40 to set1, did the set change?

**(e)** Write code to remove value 20 from set2.

**Part 2**

**1.** Create a dictionary named password\_lookup that contains usernames as keys and passwords as associated string values. Make up data for five entries.

**2.** Write a program that creates an initially empty dictionary named password\_lookup, prompting one-by-one for usernames and passwords (until a username of 'z' is read) entering each into the dictionary.

**3.** Create a dictionary named password\_hint that contains email addresses as keys, and associated values that contain both the users’ “password security question,” and the answer to the question. Make up data for dictionary entries.

**4.** Create a dictionary named member\_table that contains users’ email addresses as keys, and answers to their password hints as the associated values, and a function that generates a temporary new password and stored in the table.

**5.** Declare a set named vowels containing the strings 'a','e','i','o', 'u'. Create a function called count\_vowels that prompts the user for any English word and displays how many vowels it contains.

**6.** Create a function called word\_intersection that prompts the user for two English words, and displays which letters the two words have in common. Convert each string to a set type in order to solve this problem.

**7.** Create a function called word\_difference that prompts the user for two English words, and displays which letters are in the first word but not the second. Convert each string to a set type in order to solve this.

**8.** Create a function called get\_missing\_letters that prompts the user for two English words, and displays which letters of the alphabet are in neither of the two words. Convert each string to a set type in order to solve this.

**9.** Create a function called word\_sdifference that prompts the user for two English words, and displays which letters are in either the first word or the second word, but not both words. Convert each string to a set type in order to solve this.

**Part 3**

**1.** Write a Python function called add\_daily\_temp that is given a (possibly empty) dictionary meant to hold the average daily temperature for each day of the week, a temperature value, and the day of the week for the recorded temperature. The function should then add the temperature to the dictionary only if does not already contain a temperature for that day. The function should return the resulting dictionary, whether it is updated or not.

**2.** Write a Python function named moderate\_days that is given a dictionary containing the average daily temperature for each of the days of a week and returns a list of the days in which the average was between 70 and 79 degrees.

**3.** Write a Python function named get\_daily\_temps that prompts the user for the average temperature for each day of the week and returns a dictionary containing the information the user entered.

**4.** Write a Python function named get\_weekend\_average\_temp that is passed a dictionary of daily temperatures and returns the average temperature over the weekend for the weekly temperatures given.

**5.** Write a Python function named add\_vegetable that is passed a (possible empty) set of vegetable names and raises a ValueError exception if the given vegetable is already in the set, otherwise, add the vegetable and return a new set.

**6.** Write a Python function named num\_vowels\_consonants that is passed a string containing letters, each of which may be in either upper or lower case and returns how many vowels and consonants the string contains.

**Part 4 (Optional)**

Write a program that creates and stores student grades.

Your program should start by asking users whether they would like to add or view stored grades. If the user selects the first option, they should be repeatedly prompted for the names and module results (4CS001, 4CS015 and 4CI018) of students until the users enters a blank string, storing the information in a dictionary, before writing the data to a text file in JSON or CSV format.

If the user selects to view student grades, any data stored in the results file should be read, loaded into a dictionary and presented to the user.

Make sure to validate all inputs and utilise exception handling to avoid crashes.