

Lab 5

Dictionaries & Sets

Tasks:

- (Fill in the Missing Code) In each of the following expressions, replace the ***s with a set operator that produces the result shown in the comment. The last operation should check whether the left operand is an improper subset of the right operand. For each of the first four expressions, specify the name of the set operation that produces the specified result.
 - `{1, 2, 4, 8, 16} *** {1, 4, 16, 64, 256} # {1,2,4,8,16,64,256}`
 - `{1, 2, 4, 8, 16} *** {1, 4, 16, 64, 256} # {1,4,16}`
 - `{1, 2, 4, 8, 16} *** {1, 4, 16, 64, 256} # {2,8}`
 - `{1, 2, 4, 8, 16} *** {1, 4, 16, 64, 256} # {2,8,64,256}`
 - `{1, 2, 4, 8, 16} *** {1, 4, 16, 64, 256} # False`
- (Duplicate Email Address Removal) Write a function that receives a list of email addresses and displays only the unique addresses. Treat uppercase and lowercase letters the same. The function should use a set to get the unique email addresses from the list. Test your function with several different lists.
- Write a program that print out an alphabetical listing of all the words, and the number of times each occurs, in the text version of Alice's Adventures in Wonderland. (<https://www.gutenberg.org/files/11/11-0.txt>)
How many times does the word, alice, occur in the book?
What is the longest word in Alice in Wonderland? How many characters does it have?
- (Character Frequency Analysis) In cryptanalysis, the technique of character frequency analysis is often used to decipher a message when the key is unknown. This technique calculates the frequency of each character in the encrypted text and maps this to the frequency of characters used in, for instance, the English language. Write a script that uses a dictionary to summarize the frequency of each letter in a given text. Ignore case,

ignore spaces and assume that the user does not enter any punctuation. Display a two-column table of the letters and their frequency.

5. (Dictionary Manipulations) Using the following dictionary, which maps country names to abbreviations:

```
tlds = {'Canada': 'ca', 'United States': 'us', 'Mexico': 'mx'}
```

Write a script that shows the user a list of dictionary manipulation options to choose from. Based on the choice, the corresponding manipulation is performed, and the result is shown on the screen. The following choices can be made by the user:

- a) Show the abbreviation of a country chosen by the user.
 - b) Display all key-value pairs in a two-column format.
 - c) Add a new key-value pair to the dictionary or change the value of an existing key-value pair.
 - d) Create a new dictionary with the values of the first dictionary as keys and the keys of the first dictionary as values.
 - e) Convert all the abbreviations in the dictionary to uppercase letters.
6. (Personnel Files) Create a dictionary containing information about a company's staff. Within the dictionary, their personnel numbers are used as dictionary keys. For each staff member, their name, date of birth, and office branch are used as values. Display the dictionary's contents as a key with an indented list of staff information below it.
7. (Challenge: Writing the Word Equivalent of a Check Amount) In check-writing systems, it's crucial to prevent alteration of check amounts. One common security method requires that the amount be written in numbers and spelled out in words as well. Even if someone can alter the numerical amount of the check, it's tough to change the amount in words. Create a dictionary that maps numbers to their corresponding word equivalents. Write a script that inputs a numeric check amount that's less than 1000 and uses the dictionary to write the word equivalent of the amount. For example, the amount 112.43 should be written as

ONE HUNDRED TWELVE AND 43/100