The dictionary Junior shows a schedule for a junior year semester. The key is the course name and the value is the number of credits. Find the total number of credits taken this semester and assign it to the variable credits. Do not hardcode this – use dictionary accumulation!

Junior = {'SI 206':4, 'SI 310':4, 'BL 300':3, 'TO 313':3, 'BCOM 350':1, 'MO 300':3}

credits = 0

for c in Junior:

credits = credits + Junior[c]

print(credits)

Create a dictionary, freq, that displays each character in string str1 as the key and its frequency as the value.

str1 = "peter piper picked a peck of pickled peppers"

freq = {}

for c in str1:

if c not in freq:

freq[c] = 0

freq[c] = freq[c] + 1

print(freq)

Provided is a string saved to the variable name s1. Create a dictionary named counts that contains each letter in s1 and the number of times it occurs.

s1 = "hello"

counts = {}

for c in s1:

if c not in counts:

counts[c] = 0

counts[c] = counts[c] + 1

print(counts)

Create a dictionary, freq\_words, that displays each word in string str1 as the key and its frequency as the value.

str1 = "I wish I wish with all my heart to fly with dragons in a land apart"

keys = str1.split()

freq\_words = {}

for key in keys:

if key not in freq\_words:

freq\_words[key] = 0

freq\_words[key] = freq\_words[key] + 1

print(freq\_words)

Create a dictionary called wrd\_d from the string sent, so that the key is a word and the value is how many times you have seen that word.

sent = "Singing in the rain and playing in the rain are two entirely different situations but both can be good"

keys = sent. split ()

wrd\_d = {}

for key in keys:

if key not in wrd\_d:

wrd\_d[key] = 0

wrd\_d[key] = wrd\_d[key] + 1

print(wrd\_d)

Create the dictionary characters that shows each character from the string sally and its frequency. Then, find the most frequent letter based on the dictionary. Assign this letter to the variable best\_char.

sally = "sally sells sea shells by the sea shore"

characters = {}

for c in sally:

if c not in characters:

characters[c] = 0

characters[c] = characters[c] + 1

print(characters)

keys = list (characters. keys ())

best\_char = keys [0]

for key in keys:

if characters[key] > characters[best\_char]:

best\_char = key

print (best\_char)

Do the same as above but now find the least frequent letter. Create the dictionary characters that shows each character from string sally and its frequency. Then, find the least frequent letter in the string and assign the letter to the variable worst\_char.

sally = "sally sells sea shells by the sea shore and by the road"

characters = {}

for c in sally:

if c not in characters:

characters[c] = 0

characters[c] = characters[c] + 1

print(characters)

keys = list (characters. keys ())

worst\_char = keys [0]

for key in keys:

if characters[key] < characters[worst\_char]:

worst\_char = key

print (worst\_char)

Create a dictionary named letter\_counts that contains each letter and the number of times it occurs in string1. **Challenge:** Letters should not be counted separately as upper-case and lower-case. Instead, all of them should be counted as lower-case.

string1 = "There is a tide in the affairs of men, which taken at the flood, leads on to fortune. Omitted, all the voyage of their life is bound in shallows and in miseries. On such a full sea are we now afloat. And we must take the current when it serves, or lose our ventures."

string2 = string1.lower()

letter\_counts = {}

for c in string2:

if c not in letter\_counts:

letter\_counts[c] = 0

letter\_counts[c] = letter\_counts[c] + 1

print(letter\_counts)

Create a dictionary called low\_d that keeps track of all the characters in the string p and notes how many times each character was seen. Make sure that there are no repeats of characters as keys, such that “T” and “t” are both seen as a “t” for example.

p = "Summer is a great time to go outside. You have to be careful of the sun though because of the heat."

p1 = p. lower ()

low\_d = {}

for c in p1:

if c not in low\_d:

low\_d[c] = 0

low\_d[c] = low\_d[c] + 1

print(low\_d)