CS23336-Introduction to Python Programming

Started on	Sunday, 10 November 2024, 2:18 PM
State	Finished
Completed on	Sunday, 10 November 2024, 4:48 PM
Time taken	2 hours 29 mins
Marks	10.00/10.00
Grade	**100.00** out of 100.00

Question 1

Correct
Mark 1.00 out of 1.00
Flag question

Question text

A sentence is a list of words that are separated by a single space with no leading or trailing spaces. Each word consists of lowercase and uppercase English letters.

A sentence can be shuffled by appending the 1-indexed word position to each word then rearranging the words in the sentence.

For example, the sentence "This is a sentence" can be shuffled as "sentence4 a3 is2 This1" or "is2 sentence4 This1 a3".

Given a shuffled sentence s containing no more than 9 words, reconstruct and return the original sentence.

Example 1:

Input:

is2 sentence4 This1 a3

Output:

This is a sentence

Explanation: Sort the words in s to their original positions "This1 is2 a3 sentence4", then remove the numbers.

Example 2:

Input:

Myself2 Me1 I4 and3

Output:

Me Myself and I

Explanation: Sort the words in s to their original positions "Me1 Myself2 and3 I4", then remove the numbers.

Constraints:

2 <= s.length <= 200

s consists of lowercase and uppercase English letters, spaces, and digits from 1 to 9.

The number of words in s is between 1 and 9.

The words in s are separated by a single space.

s contains no leading or trailing spaces.

Answer:(penalty regime: 0 %)

```
def fun():
    s=input().split()

a=sorted(s,key=lamb
da
s1:int(''.join(filter(str.is
digit,s1))))
    x='
'.join(''.join(filter(str.is
alpha,s1))for s1 in a)
    return x
print(fun())
```

Feedback

Input	Expected	Got
is2 sentence4 This1 a3	This is a sentence	This is a sentence

Passed all tests!

Correct
Marks for this submission: 1.00/1.00.

Question 2

Correct
Mark 1.00 out of 1.00
☐ Flag question

Question text

A company wants to send its quotation secretly to its client. The company decided to encrypt the amount they are sending to their client with some special symbols so that the equation amount will not be revealed to any external person. They used the special symbols !,@,#,$,%,^,&,*,>,< for 0,1,2,3,4,5,6,7,8,9 respectively. Write a python code to help the company to convert the amount to special symbols.

(Value rounded off to 2 decimal points)

Input

n: Float data type which reads amount to send

Output

s: : String data type which displays symbols

Sample Testcase 1

Input

10000

Output

@!!!!.!!

Sample Testcase2

1234.56

Output

@#$%.^&

For example:

Input	Result
1345.23	@$%^.#$
15000.59	@^!!!.^<
156789	@^&*><.!!

Answer:(penalty regime: 0 %)

```
a=
{'!':'0','@':'1','#':'2','$'
:'3','%':'4','^':'5','&':'6'
,'*':'7','>':'8','<':'9','.':'
.'}
b=float(input())
c=f"{b:.2f}"
d=str(c)
f=''
for i in d:
    for char, val in
a.items():
        if i in val:
            f=f+char
if len(d)==len(f):
    print(f)
```

Feedback

Input	Expected	Got
1345.23	@$%^.#$	@$%^.#$
15000.59	@^!!!.^<	@^!!!.^<
1234	@#$%.!!	@#$%.!!
156789	@^&*><.!!	@^&*><.!!

Passed all tests!

Correct
Marks for this submission: 1.00/1.00.

Question 3

Correct
Mark 1.00 out of 1.00
☐ Flag question

Question text

Give a dictionary with value lists, sort the keys by summation of values in value list.

Input : test_dict = {'Gfg' : [6, 7, 4], 'best' : [7, 6, 5]}

Output : {'Gfg': 17, 'best': 18}

Explanation : Sorted by sum, and replaced.

Input : test_dict = {'Gfg' : [8,8], 'best' : [5,5]}

Output : {'best': 10, 'Gfg': 16}

Explanation : Sorted by sum, and replaced.

Sample Input:

2

Gfg 6 7 4

Best 7 6 5

Sample Output

Gfg 17

Best 18

For example:

Input	Result
2 Gfg 6 7 4 Best 7 6 5	Gfg 17 Best 18

Answer:(penalty regime: 0 %)

```
n=int(input())
data={}
for i in range(n):
    en=input().split()
    key=en[0]
    val =[int(num) for
num in en[1:]]
    data[key]=val
sum1={}
for key in data:

sum1[key]=sum(dat
a[key])
s=sorted(sum1.item
s(),key=lambda
item:item[1])
for item in s:
```

Feedback

Input	Expected	Got
2 Gfg 6 7 4 Best 7 6 5	Gfg 17 Best 18	Gfg 17 Best 18
2 Gfg 6 6 Best 5 5	Best 10 Gfg 12	Best 10 Gfg 12

Passed all tests!

Correct
Marks for this submission: 1.00/1.00.

Question 4

Correct
Mark 1.00 out of 1.00
☐ Flag question

Question text

Objective:

Develop a Python program that takes an input string from the user and counts the number of occurrences of each vowel (a, e, i, o, u) in the string. The program should be case-insensitive, meaning it should treat uppercase and lowercase vowels as the same.

Description:

Vowels play a significant role in the English language and other alphabet-based languages. Counting vowels in a given string is a fundamental task that can be applied in various text processing applications, including speech recognition, linguistic research, and text analysis. The objective of this problem is to create a Python script that accurately counts and displays the number of times each vowel appears in a user-provided string.

Program Requirements:

Input:

First line reading String as input, The string can contain any characters, including letters, numbers, and special characters.

Output:

Display the number of occurrences of each vowel in the string.

The output should list each vowel followed by its count.

Example:

Consider the following example for better understanding:

- **Input:** "Python Programming"
- **Output**

  a = 1
  e = 0
  i = 1
  o = 2
  u = 0

For example:

Input	Result
Hello World	a = 0 e = 1 i = 0 o = 2 u = 0
Python	a = 0 e = 0 i = 0 o = 1 u = 0

Answer:(penalty regime: 0 %)

```
a=input().lower()
c=0
for i in 'aeiou':
    c=0
    if i in a:
        c=a.count(i)
        print("{} =
{}".format(i,c))
    else:
        print("{} =
{}".format(i,c))
```

Feedback

Input	Expected	Got
Hello World	a = 0 e = 1 i = 0 o = 2 u = 0	a = 0 e = 1 i = 0 o = 2 u = 0
AEIOU aeio u	a = 2 e = 2 i = 2 o = 2 u = 2	a = 2 e = 2 i = 2 o = 2 u = 2
Python	a = 0 e = 0 i = 0 o = 1 u = 0	a = 0 e = 0 i = 0 o = 1 u = 0
abcdefghijklmnopqrstuvwxyz	a = 1 e = 1 i = 1 o = 1 u = 1	a = 1 e = 1 i = 1 o = 1 u = 1
12345!@#$%AEIOU	a = 1 e = 1 i = 1 o = 1 u = 1	a = 1 e = 1 i = 1 o = 1 u = 1

Passed all tests!

Correct
Marks for this submission: 1.00/1.00.

Question 5

Correct
Mark 1.00 out of 1.00
☐ Flag question

Question text

A sentence is a string of single-space separated words where each word consists only of lowercase letters.A word is uncommon if it appears exactly once in one of the sentences, and does not appear in the other sentence.

Given two sentences s1 and s2, return a list of all the uncommon words. You may return the answer in any order.

Example 1:

Input: s1 = "this apple is sweet", s2 = "this apple is sour"

Output: ["sweet","sour"]

Example 2:

Input: s1 = "apple apple", s2 = "banana"

Output: ["banana"]

Constraints:

1 <= s1.length, s2.length <= 200

s1 and s2 consist of lowercase English letters and spaces.

s1 and s2 do not have leading or trailing spaces.

All the words in s1 and s2 are separated by a single space.

Note:

Use dictionary to solve the problem

For example:

Input	Result
this apple is sweet this apple is sour	sweet sour

Answer:(penalty regime: 0 %)

```
from collections
import Counter
def fun(s1,s2):

words=s1.split()+s2.s
plit()
    wc=Counter(words)
    return [word for
word,count in
wc.items() if
count==1]

s1=input()
s2=input()
print(*fun(s1,s2))
```

Feedback

Input	Expected	Got
this apple is sweet this apple is sour	sweet sour	sweet sour
apple apple banana	banana	banana

Passed all tests!

Correct
Marks for this submission: 1.00/1.00.

Question 6

Correct
Mark 1.00 out of 1.00
☐ Flag question

Question text

In the game of Scrabble™, each letter has points associated with it. The total score of a word is the sum of the scores of its letters. More common letters are worth fewer points while less common letters are worth more points. The points associated with each letter are shown below:

Points Letters

1 A, E, I, L, N, O, R, S, T and U

2 D and G

3 B, C, M and P

4 F, H, V, W and Y

5 K

8 J and X

10 Q and Z

Write a program that computes and displays the Scrabble™ score for a word. Create a dictionary that maps from letters to point values. Then use the dictionary to compute the score.

A Scrabble™ board includes some squares that multiply the value of a letter or the value of an entire word. We will ignore these squares in this exercise.

Sample Input

REC

Sample Output

REC is worth 5 points.

For example:

Input	Result
REC	REC is worth 5 points.

Answer:(penalty regime: 0 %)

```
a={1:
['A','E','I','L','N','O','R','
S','T','U'],2:['D','G'],3:
['B','C','M','P'],4:
['F','H','V','W','Y'],5:'K',
8:['J','X'],10:['Q','Z']}
def ls(l):
    for p,x in a.items():
        if l in x:
            return p
    return 0
b=input().upper()
s=0
for i in b:
    s=s+ls(i)
print(f'{b} is worth
{s} points.')
```

Feedback

Input	Expected	Got
GOD	GOD is worth 5 points.	GOD is worth 5 points.
REC	REC is worth 5 points.	REC is worth 5 points.

Passed all tests!

Correct
Marks for this submission: 1.00/1.00.

Question 7

Correct
Mark 1.00 out of 1.00
☐ Flag question

Question text

Given a number, convert it into corresponding alphabet.

Input	Output

1	A
26	Z
27	AA
676	YZ

Input Format

Input is an integer

Output Format

Print the alphabets

Constraints

1 <= num <= 4294967295

Sample Input 1

26

Sample Output 1

Z

For example:

Test	Result
`print(excelNumber(26))`	Z

Answer:(penalty regime: 0 %)

Reset answer