Question: 1

```
def count_letter(word, search):
        return count
word = input("Enter the words to search: ").lower()
search = input("Enter the character to search: ").lower()
count = 0
for char in word:
    if char == search:
        count += 1
print(count_letter(word, search))
```

Problem Solving Using Python and R Lab

Lab4. Python String Processing

Question1. Develop a function **count_letter(string, search)** that returns the number of times search character appears in a string.

Test cases:

1. Str = "hello world". Search = 'o'. Calling ***count_letter(str, search)*** should return output 2
2. Str = "HeLlo wOrld". Search = 'o'. Then, **calling count_letter(str, search)** will return output 1

Modify **count_letter()** so that it ignores case sensitivity, so that **o** and **O** are same.

3. Str = "HeLlo wOrld". Search = 'o'. Calling **count_letter(str, search)** will return output 2

Question:2

```
s = input ("Enter the string: ")
a = s.lower()
vowels = "aeiou"
consonants = "bcdfghjklmnpqrstvwxyz"
digits = "1234567890"

c = 0
v = 0
d = 0
sp = 0

for i in a:
    if i in vowels:
        v+=1
    elif i in consonants:
        c+=1
    elif iin digits:
        d+=1
    elif i in space:
        sp+=1
print("\n Space: ", sp, "\n Digits: ", d, "\n Vowels: ", v,
                "\n Consonants: ", c)
```

Question2. Write a program that counts the number of spaces, digits, vowels and consonants in a string that the user inputs. Print the string, no of spaces, no of digits, no of vowels and no of consonants.

Test case: Enter a string: **Bishop Heber College 17**. Then output should be:

Given string: Bishop Heber College 17
No. of spaces: 3
No. of digits: 2
No. of vowels: 7
No. of consonants: 12

Question3. Develop a function **remove_punctuation(str)** that returns the string after removing the following punctuations.

Punctuation List = "!\"#$%&'()*+,-./:;<=>?@[\]^`{|}~"

Test cases:
1. Str = "Bishop's College !....." . Calling **remove_punctuation(str)** should return output as "Bishops College"
2. Str = "#bhc trending @cs $placements::>." Calling **remove_punctuation(str)** should return output as "bhc trending cs placements"

```
Question3:  def remove_punctuation(string):
                punctuations = '''!()-[];:'"\,<>,./?@#$%^&*_'''
                for x in string.lower():
                    if x in punctuations:
                        string = string.replace(x, "")
                print(string)
            string = (input("Enter the string: "))
            remove_punctuation(string)
```

Question: 4

```
word = input("Enter a word to translate to pig latin:")
def piglatin(word):
    ay = 'ay'
    way = 'way'
    p = '-'
    consonant = ('B', 'C', 'D', 'F', 'G', 'H', 'J', 'K', 'L', 'M', 'N',
      'P', 'Q', 'R', 'S', 'T', 'Y', 'V', 'X', 'Z')
    vowel = ('A', 'E', 'I', 'O', 'U')
    first_letter = word[0]
    first_letter = str(first_letter)
    first-letter = first_letter.upper()
    if first_letter in consonant:
        print(first_letter, 'is a consonant')
        length_of_word = len(word)
        remove_first_letter = word[1: length_of_word]
        pig_latin = remove-first-letter + p + first_letter
                    + ay
        print("the word in pig Latin is:", pig_latin)
    else first_letter in vowel:
        print(first_letter, "is a vowel")
        pig_latin = word + p + way
        print("theword in pig latin is:", pig_latin)
    else:
        print("I dont know what", first_letter, 'is')
```

Question4. Write a program that asks the user for a word. Translate their word into Pig Latin. Pig Latin game takes the first consonant (or set of first consonants) of an English word, moves it to the end of the word and suffixes an **ay**. If the first letter is a vowel, do not move that vowel, but instead add "**way**" at the end of the word.

Test Cases:

1. Enter a word: **pig**
   Output: **ig-pay**
2. Enter a word: **banana**
   Output: **anana-bay**
3. Enter a word: **trash**
   Output: **ash-tray**
4. Enter a word: **apple**
   Output: **apple-way**
5. Enter a word: **orange**
   Output: **orange-way**

Modify your program so that it becomes a function **piglatin(word)** and returns translated word as output. Call this function 3 times with the same inputs and validate the outputs.

Question:4

piglatin(word)