ODIN SCHOOL_ Python Project 101 Submission_ Mandatory Project Submissions

Q1: You are working in a bank, and you have been given two lists of the employees who worked in 2021. Employees' names in list 1 are Ramesh, Suresh, Mahesh, Ali, Jacob, and Saritha. List 2 contains the names of Ali, Mukesh, Mahesh, Jacob, Sai, and Sarita. Please write a program that helps to identify people who are common in both lists. Please do not use any in-built function.

```
def find_common_employees(list1, list2):
    common_employees = []

    for employee1 in list1:
        for employee2 in list2:
            if employee1 == employee2:
                common_employees.append(employee1)
                break

    return common_employees

# Given employee lists
list1 = ["Ramesh", "Suresh", "Mahesh", "Ali", "Jacob", "Saritha"]
list2 = ["Ali", "Mukesh", "Mahesh", "Jacob", "Sai", "Sarita"]

common_employees = find_common_employees(list1, list2)
print("Common employees:", common_employees)
```

output: Common employees: ['Mahesh', 'Ali', 'Jacob']

Q2: While entering data, someone entered a few names as a common string "Ramesh Suresh Mohit". Please write a program which separates all the names and converts them into a list. Once converted into a list, please write a program that adds their age.

Ramesh: 25

Suresh: 22

Mohit: 26

```
# Dictionary containing names as keys and ages as values
ages = {
    "Ramesh": 25,
    "Suresh": 22,
    "Mohit": 26
}

# Common string of names
common_names_string = "Ramesh Suresh Mohit"

# Split the common string into a list of names
```

```
common_names_list = common_names_string.split()

# Initialize total age to 0
total_age = 0

# Iterate through the list of names and add their ages
for name in common_names_list:
    if name in ages:
        total_age += ages[name]

# Print the total age
print("Total age:", total_age)
```

output:Total age: 73

Q3: You are working in a medical store. A patient came to your medical store and asked to buy 2 strips of paracetamol, 3 strips of azithromycin, and 5 strips of Vitamin C. One strip of paracetamol costs Rs 35, one strip of azithromycin costs Rs 49, and one strip of vitamin c costs Rs. 33. Patient gave you Rs 2000. Please tell us what is the total cost of each medicine, the total cost of all medicine, and how much money you refunded to the patient.

```
# Prices of each medicine
price_paracetamol = 35
price_azithromycin = 49
price_vitamin_c = 33

# Quantities requested by the patient
quantity_paracetamol = 2
quantity_azithromycin = 3
quantity_vitamin_c = 5

# Total cost of each medicine
total_cost_paracetamol = price_paracetamol * quantity_paracetamol
total_cost_azithromycin = price_azithromycin * quantity_azithromycin
total_cost_vitamin_c = price_vitamin_c * quantity_vitamin_c

# Total cost of all medicine
total_cost_all_medicine = total_cost_paracetamol +
total_cost_azithromycin + total_cost_vitamin_c

# Amount paid by the patient
amount_paid = 2000

# Calculate the refund amount
refund_amount = amount_paid - total_cost_all_medicine

# Print the results
```

```
print("Total cost of Paracetamol:", total_cost_paracetamol)
print("Total cost of Azithromycin:", total_cost_azithromycin)
print("Total cost of Vitamin C:", total_cost_vitamin_c)
print("Total cost of all medicine:", total_cost_all_medicine)
print("Refund amount:", refund_amount)
```

output:
Total cost of Paracetamol: 70
Total cost of Azithromycin: 147
Total cost of Vitamin C: 165
Total cost of all medicine: 382
Refund amount: 1618

Q4: Accept a sentence as input and find the number of vowels in it. Assume that the sentence has no punctuation marks. For example, I am learning python contains 6 vowels. This function should be applicable for all other different sentences.

```
def count_vowels(sentence):
    vowels = "aeiouAEIOU"  # List of vowel characters
    vowel_count = 0

    for char in sentence:
        if char in vowels:
            vowel_count += 1

    return vowel_count

# Accept a sentence as input
input_sentence = input("Enter a sentence: ")

# Calculate the number of vowels
num_vowels = count_vowels(input_sentence)

# Print the result
print("Number of vowels:", num_vowels)
```

```
output: Enter a sentence: a
Number of vowels: 1
```

Q5: You have been appointed by the election commission to create a website. Your first task is to work on a program which tells candidates if they are eligible for voting or not. If they are eligible, your output should be ‘Congrats! You are eligible’; otherwise, it should tell that you have to return after X number of years. The eligibility criteria for voting is 18 years.

For example, If someone is 18 or above, your output should be ‘Congrats! You are eligible’. If someone’s age is 15 years, it should print output as ‘return after 3 years’.

```python
def check_voting_eligibility(age):
    if age >= 18:
        return "Congrats! You are eligible"
    else:
        years_to_wait = 18 - age
        return f"Return after {years_to_wait} years"

# Accept candidate's age as input
candidate_age = int(input("Enter your age: "))

# Check eligibility and print the result
result = check_voting_eligibility(candidate_age)
print(result)
```

output: Enter your age: 32
Congrats! You are eligible

Q6: Given a list of integers, find the cumulative sum of the elements of the list and store them in another list.

A = [1, 2, 3, 4, 5]

Output:

[1, 3, 6, 10, 15]

```python
def calculate_cumulative_sum(input_list):
    cumulative_sum = []
    total = 0

    for num in input_list:
        total += num
        cumulative_sum.append(total)

    return cumulative_sum

# Given list of integers
A = [1, 2, 3, 4, 5]

# Calculate cumulative sum
cumulative_result = calculate_cumulative_sum(A)

# Print the output
print(cumulative_result)
```

output: [1, 3, 6, 10, 15]

Q7:WAP to encode a message entered by user as per below conditions:

for every word in the sentence,

1. If the word starts with a vowel, encode it as the first and last letter of the word.
2. If the word starts with a consonant, remove all vowels from it.

Ensure the case insensitive comparisons/checks are performed.

Please enter your text:The quick brown fox used to sleep inside this box

Encoded Msg: Th qck brwn fx ud t slp ie ths bx

```
def encode_word(word):
    vowels = "aeiouAEIOU"

    if word[0] in vowels:
        return word[0] + word[-1]
    else:
        return ''.join(char for char in word if char.lower() not in
vowels)

# Accept the user's input
user_text = input("Please enter your text: ")

# Split the input text into words
words = user_text.split()

# Encode each word and create the encoded message
encoded_message = [encode_word(word) for word in words]

# Join the encoded words to form the final message
encoded_msg = ' '.join(encoded_message)

# Print the encoded message
print("Encoded Msg:", encoded_msg)
```

output: Please enter your text: hello my name is shruthi

Encoded Msg: hll my nm is shrth

Q8:Write a program to implement run length encoding of a string

RLE: Consecutive repetition of a character has to be replaced with the count of occurrences and the character.

Enter your string :aabbbccdddae

Encoded: 2a3b2c3d1a1e

```python
def run_length_encode(input_string):
    encoded_string = ""
    count = 1

    for i in range(1, len(input_string)):
        if input_string[i] == input_string[i - 1]:
            count += 1
        else:
            encoded_string += str(count) + input_string[i - 1]
            count = 1

    encoded_string += str(count) + input_string[-1]  # Handle the last
character

    return encoded_string

# Accept user input
user_string = input("Enter your string: ")

# Encode the string using run-length encoding
encoded_result = run_length_encode(user_string)

# Print the encoded result
print("Encoded:", encoded_result)
```

output: Enter your string: "shruthi"
Encoded: 1"1s1h1r1u1t1h1i1"