Title: Mastering Advanced Python: Tricky and Situational Programming

1. Write a function to count the number of vowels and consonants in a given string.

In [ ]:
```
def count_vowels_consonants():
  text = input("Enter a string:")
  vowels = "aeiouAEIOU"           #List of all vowels(both lower and upper )
  vowel_count = 0
  consonant_count = 0

  for char in text:      #Loop through each character in the string
    # Check if the character is an alphabet (ignores numbers, spaces, etc.)
    if char.isalpha():
      if char in vowels:
        vowel_count += 1
      else:
        consonant_count += 1

  print("Number of vowels:", vowel_count)
  print("Number of consonants:", consonant_count)

#calling the function
count_vowels_consonants()
```

```
Enter a string:happy nEW year 2025
Number of vowels: 4
Number of consonants: 8
```

2. Write a function to return the minimum and maximum elements from a tuple.

In [ ]:
```
#Function to find min and max in a tuple
def find_min_max(my_tuple):
  minimum = min(my_tuple)
  maximum = max(my_tuple)
  # return both values as a tuple
  return minimum,maximum


# Taking input from users
user_input = input("Enter the number seperated by commas:")

# Convert input string to tuple of integers
user_tuple = tuple(map(int, user_input.split(",")))

# call the function
min_val, max_val = find_min_max(user_tuple)

# Result
print("Minimum value:", min_val)
print("Maximum value:", max_val)
```

```
Enter the number seperated by commas:256,299,444,67,79
Minimum value: 67
Maximum value: 444
```

3. Write code to unpack the first two elements and the rest from a tuple of unknown length.

In [ ]:
```
# suppose we have tuple
my_tuple = (10,20,30,40,50)

# unpack the first two elements and collecting the rest
first, second, *rest = my_tuple

# Displaying the result
print("First element:", first)
print("Second element:", second)
print("Rest of the elements:", rest)
```

```
First element: 10
Second element: 20
Rest of the elements: [30, 40, 50]
```

4. Write a function that takes a string and returns the frequency of each word.

In [2]:
```
def word_frequency(text):
  words = text.split()   # split the string into words
  frequency = {}


  for word in words:
    word = word.lower()
    if word in frequency:
      frequency[word] += 1
    else:
      frequency[word] = 1

  return frequency

# taking input from the users
user_input = input("Enter a sentance:")

#call the function
result = word_frequency(user_input)


# Displaying the result
print("Word frequency:", result)
```

```
Word frequency: {'python': 2, 'is': 2, 'fun': 1, 'and': 1, 'powerful': 1}
```

5. Write a function to find duplicate elements in a list.

In [ ]:
```
def find_duplicate(input_list):
  seen = set()  #stores unique elements
  duplicate = set() #stores element that appear more than once

  for item in input_list:
    if item in seen:
      duplicate.add(item)
    else:
      seen.add(item)
  return list(duplicate)
```

```
#Predefined input list
my_list = [1, 2, 3, 4, 2, 3, 5, 6, 1]

#call the function
result = find_duplicate(my_list)

#show the result
print("Duplicate elements:", result)
```

```
Duplicate elements: [1, 2, 3]
```

6. Use a filter with a lambda function to extract even numbers from a list.

In [ ]:
```
# Sample list of numbers
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

# Use filter() with a lambda function to get even numbers
even_numbers = list(         #convert the result to a list
                    filter(  #Use filter to find matching values
                           lambda x: x % 2 == 0, #keep only numbers where x % 2
                           numbers          ## The list to filter

                    )
    )
print(even_numbers)
```

```
[2, 4, 6, 8, 10]
```

7. Create a Python module named mymath.py with functions for addition, subtraction, and multiplication. Import and use it.

In [1]:
```
'''
This code for file.
def add(a,b):
    return a + b

def subtract(a,b):
    return a-b

def multiply(a,b):
    return a*b
'''


import mymath

print(mymath.add(2, 3))
print(mymath.subtract(10, 4))
print(mymath.multiply(6, 7))
```

```
5
6
42
```

8. Sort a List of Tuples Based on the Second Element

In [ ]:
```
# sample list of tuple
tuple_list = [(1, 2), (4, 3), (2, 6), (5, 1)]

# Sort based on the second element (index 1 of each tuple)
```

```
sorted_list = sorted(tuple_list, key=lambda x: x[1])
print(sorted_list)
```

```
[(5, 1), (1, 2), (4, 3), (2, 6)]
```

9. Write a function to find the symmetric difference (elements not common) between two sets.

In [ ]:
```
def symmetric_difference(set1,set2):
  return set1.symmetric_difference(set2)

#Sample set
a = {1, 2, 3, 4}
b = {3, 4, 5, 6}

result = symmetric_difference(a,b)
print("Symmetric Difference:",result)
```

```
Symmetric Difference: {1, 2, 5, 6}
```

10.Write a function that returns a list of prime numbers up to a given number n.

In [6]:
```
# funtion to check if number is a prime
is_prime = lambda x: x > 1 and all(x % i != 0 for i in range(2, int(x**0.5)+1))

# Take input from user
try:
    n = int(input("Enter a number:"))
    prime_list = list(filter(is_prime, range(2,n+1)))
    print("Prime number up to ", n, "are", prime_list)

except Valueerror:
    print("Please enter a valid number")
```

```
Enter a number:20
Prime number up to  20 are [2, 3, 5, 7, 11, 13, 17, 19]
```

In [ ]:
```

```