**DATE: 19.06.2025**

**Python Coding Challenge**

Topic: List, Tuple, Dictionary, Set | Total Questions: 10 | Time: 60 minutes

Section A: List (3 Questions):

Section B: Tuple (2 Questions):

Section C: Dictionary (3 Questions):

Section D: Set (2 Questions):

**Q1**. Write a Python program to remove all duplicates from a list without using the set() function.

Input Example: [1, 2, 2, 3, 4, 4, 5]

Output: [1, 2, 3, 4, 5]

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

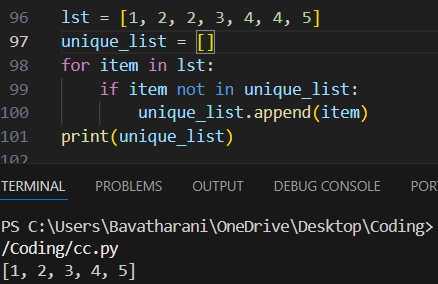
unique\_list = []

for item in lst:

    if item not in unique\_list:

        unique\_list.append(item)

print(unique\_list)



**Q2**. Given a list of integers, write a program to find the second highest unique number.

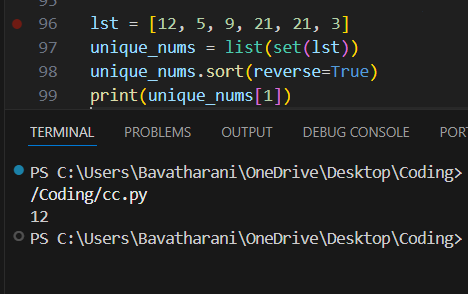
Input Example: [12, 5, 9, 21, 21, 3]

Output: 12

lst = [12, 5, 9, 21, 21, 3]

unique\_nums = list(set(lst))

unique\_nums.sort(reverse=True)

print(unique\_nums[1]) 

**Q3.** Rotate a list to the right by k positions.

Input: List = [1, 2, 3, 4, 5], k = 2

Output: [4, 5, 1, 2, 3]

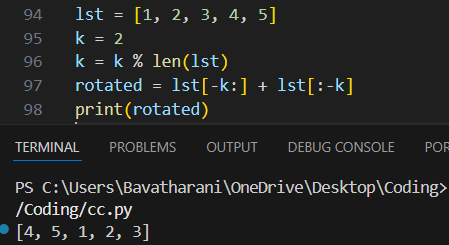
lst = [1, 2, 3, 4, 5]

k = 2

k = k % len(lst)

rotated = lst[-k:] + lst[:-k]

print(rotated)



**Q4**. Write a Python program to multiply the elements of each tuple in a list of tuples and return a new list.

Input: [(2, 4), (3, 5), (4, 6)]

Output: [8, 15, 24]

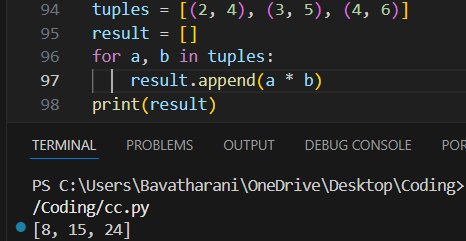
tuples = [(2, 4), (3, 5), (4, 6)]

result = []

for a, b in tuples:

result.append(a \* b)

print(result)



**Q5.** Given a tuple of integers, write a program to count how many times each element occurs.

Input: (1, 2, 2, 3, 1, 4, 2)

Output: {1: 2, 2: 3, 3: 1, 4: 1}

tpl = (1, 2, 2, 3, 1, 4, 2)

freq = {}

for item in tpl:

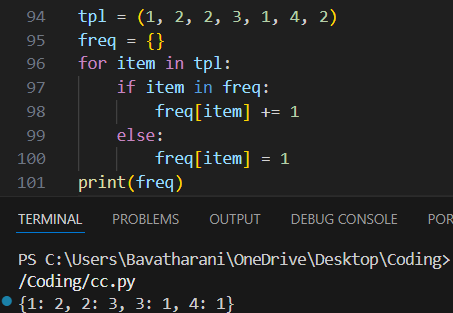
if item in freq:

freq[item] += 1

else:

freq[item] = 1

print(freq)



**Q6**. Write a Python program to count the frequency of each character in a string using a dictionary.

Input: 'banana'

Output: {'b': 1, 'a': 3, 'n': 2}

s = 'banana'

freq = {}

for ch in s:

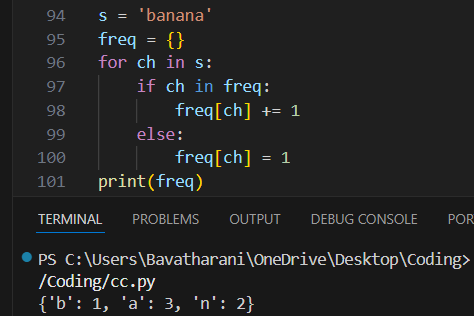
if ch in freq:

freq[ch] += 1

else:

freq[ch] = 1

print(freq)



**Q7**. Merge two dictionaries such that common keys have their values summed.

Input: {'apple': 10, 'banana': 5}, {'banana': 3, 'orange': 7}

Output: {'apple': 10, 'banana': 8, 'orange': 7}

d1 = {'apple': 10, 'banana': 5}

d2 = {'banana': 3, 'orange': 7}

merged = d1.copy()

for key in d2:

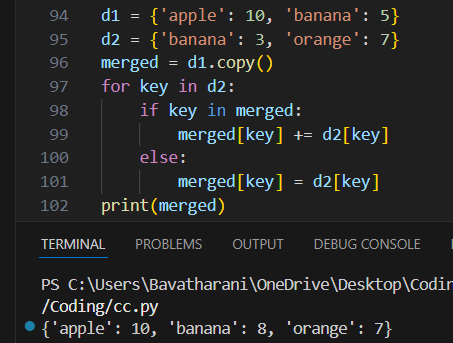
if key in merged:

merged[key] += d2[key]

else:

merged[key] = d2[key]

print(merged)



**Q8**. Given a dictionary of student names and their marks, print the name(s) of the student(s) with the highest marks.

Input: {'Alice': 85, 'Bob': 92, 'Carol': 92}

Output: ['Bob', 'Carol']

students = {'Alice': 85, 'Bob': 92, 'Carol': 92}

max\_mark = max(students.values())

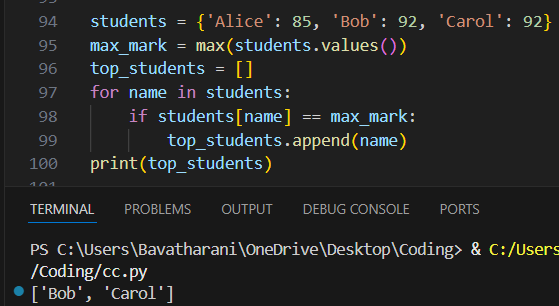
top\_students = []

for name in students:

if students[name] == max\_mark:

top\_students.append(name)

print(top\_students)



**Q9**. Write a Python program to find all common elements among three lists using set operations.

Input: [1, 2, 3], [2, 3, 4], [3, 2, 5]

Output: {2, 3}

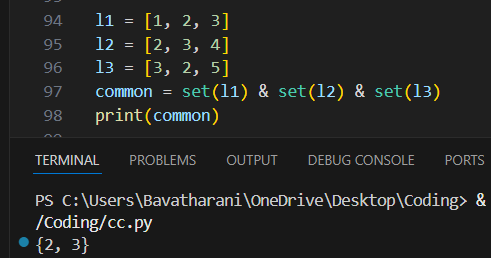
l1 = [1, 2, 3]

l2 = [2, 3, 4]

l3 = [3, 2, 5]

common = set(l1) & set(l2) & set(l3)

print(common)



**Q10**. From a sentence entered by the user, extract and display all unique words using a set.

Input: 'this is a test this is fun'

Output: {'this', 'is', 'a', 'test', 'fun'}

sentence = 'this is a test this is fun'

words = sentence.split()

unique\_words = set(words)

print(unique\_words)

