Rajalakshmi Engineering College

Name: Tharun Sathishkumar
Email: 240701563@rajalakshmi.edu.in
Roll no: 240701563
Phone: 9363661870
Branch: REC
Department: I CSE FF
Batch: 2028
Degree: B.E - CSE

Scan to verify results

NeoColab_REC_CS23221_Python Programming

REC_Python_Week 4_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 38.5

Section 1 : Coding

1. Problem Statement

Meena is analyzing a list of integers and needs to count how many numbers in the list are even and how many are odd. She decides to use lambda functions to filter the even and odd numbers from the list.

Write a program that takes a list of integers, counts the number of even and odd numbers using lambda functions, and prints the results.

Input Format

The first line contains an integer n, representing the number of integers in the list.

The second line contains n space-separated integers.

Output Format

The first line of output prints an integer representing the count of even numbers.

The second line of output prints an integer representing the count of odd numbers.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 7
12 34 56 78 98 65 23

Output: 5
2

Answer

```
def count_even_odd():
    n = int(input().strip())
    numbers = list(map(int, input().strip().split()))
    even_count = len(list(filter(lambda x: x % 2 == 0, numbers)))
    odd_count = len(list(filter(lambda x: x % 2 != 0, numbers)))


    print(even_count)
    print(odd_count)

if __name__ == "__main__":
    count_even_odd()
```

Status : Correct ***Marks : 10/10***

2. Problem Statement

You are tasked with designing a shipping cost calculator program that calculates the shipping cost for packages based on their weight and destination. The program utilizes different shipping rates for domestic, international, and remote destinations. The rates for each destination type are provided as global constants.

Constant Values:

DOMESTIC_RATE = 5.0

INTERNATIONAL_RATE = 10.0

REMOTE_RATE = 15.0

Function Signature: calculate_shipping(weight, destination)

Formula: shipping cost = weight * destination rate

Input Format

The first line of the input consists of a float representing the weight of the package.

The second line consists of a string representing the destinations(Domestic or International or Remote).

Output Format

The program outputs any one of the following:

1. If the input is valid and the destination is recognized, the output should consist of a single line stating the calculated shipping cost for the given weight and destination in the format: "Shipping cost to [destination] for a [weight] kg package: $[calculated cost]" with two decimal places.
2. If the input weight is not a positive float, print "Invalid weight. Weight must be greater than 0."
3. If the input destination is not one of the valid options, print "Invalid destination."

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 5.5
Domestic
Output: Shipping cost to Domestic for a 5.5 kg package: $27.50

Answer

```
#

DOMESTIC_RATE = 5.0
INTERNATIONAL_RATE = 10.0
REMOTE_RATE = 15.0

def calculate_shipping(weight, destination):
   dest_lower = destination.strip().lower()
   if dest_lower == "domestic":
      rate = DOMESTIC_RATE
      destination_name = "Domestic"
   elif dest_lower == "international":
      rate = INTERNATIONAL_RATE
      destination_name = "International"
   elif dest_lower == "remote":
      rate = REMOTE_RATE
      destination_name = "Remote"
   else:
      return None, None
   return weight * rate, destination_name

try:
   weight = float(input())
   destination = input()

   if weight <= 0:
      print("Invalid weight. Weight must be greater than 0.")
   else:
      shipping_cost, destination_name = calculate_shipping(weight, destination)
      #if shipping_cost is not None:
         #print(f"Shipping cost to {destination_name} for a {weight} kg package:
${shipping_cost:.2f}")
      #else:
         #print("Invalid destination.")
except ValueError:
   print("Invalid input. Please enter a valid number for weight.")

if shipping_cost is not None:
   print(f"Shipping cost to {destination} for a {weight} kg package:
${shipping_cost:.2f}")
```

Status : Partially correct ***Marks : 8.5/10***

3. Problem Statement

Arjun is working on a mathematical tool to manipulate lists of numbers. He needs a program that reads a list of integers and generates two lists: one containing the squares of the input numbers, and another containing the cubes. Arjun wants to use lambda functions for both tasks.

Write a program that computes the square and cube of each number in the input list using lambda functions.

Input Format

The input consists of a single line of space-separated integers representing the list of input numbers.

Output Format

The first line contains a list of the squared values of the input numbers.

The second line contains a list of the cubed values of the input numbers.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 1 2 3
Output: [1, 4, 9]
[1, 8, 27]

Answer

```
def compute_squares_and_cubes():

    numbers = list(map(int, input().strip().split()))


    squares = list(map(lambda x: x ** 2, numbers))
    cubes = list(map(lambda x: x ** 3, numbers))
    print(squares)
    print(cubes)


if __name__ == "__main__":
    compute_squares_and_cubes()
```

Status : Correct ***Marks : 10/10***

4. Problem Statement

Develop a text analysis tool that needs to count the occurrences of a specific substring within a given text string.

Write a function count_substrings(text, substring) that takes two inputs: the text string and the substring to be counted. The function should count how many times the substring appears in the text string and return the count.

Function Signature: count_substrings(text, substring)

Input Format

The first line of the input consists of a string representing the text.

The second line consists of a string representing the substring.

Output Format

The output should display a single line of output containing the count of occurrences of the substring in the text string.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: programming is fun and programming is cool
programming
Output: The substring 'programming' appears 2 times in the text.

Answer

```
def count_substrings(text, substring):

    count = text.count(substring)
    return count


if __name__ == "__main__":
    text = input().strip()
    substring = input().strip()
    count = count_substrings(text, substring)
    print(f"The substring '{substring}' appears {count} times in the text.")
```

Status : Correct ***Marks : 10/10***