

Rajalakshmi Engineering College

Name: Jenell S G
Email: 240701212@rajalakshmi.edu.in
Roll no: 2116240701212
Phone: 7418493255
Branch: REC
Department: I CSE AH
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 : 40

Section 1 : Coding

1. 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):
```

```
    if weight<=0 or weight>1000.0:
```

```
        print("Invalid weight. Weight must be greater than 0.")
```

```
    return None
```

```
if len(destination)<1 or len(destination)>15:  
    print("Invalid destination.")  
    return None
```

```
if destination == "Domestic":  
    rate = DOMESTIC_RATE  
elif destination == "International":  
    rate=INTERNATIONAL_RATE  
elif destination == "Remote":  
    rate = REMOTE_RATE  
else:  
    print("Invalid destination.")  
    return None
```

```
    return weight*rate  
weight = float(input())  
destination=input()
```

```
shipping_cost = calculate_shipping(weight,destination)
```

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

Status : Correct

Marks : 10/10

2. 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

```
# You are using Python
```

```
# Read input
```

```
n = int(input())
```

```
numbers = list(map(int, input().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)
```

Status : Correct

Marks : 10/10

3. Problem Statement

Amrita is developing a password strength checker for her website. She wants the checker to consider the length and the diversity of characters used in the password. A strong password should be long and include a mix

of character types: uppercase, lowercase, digits, and special symbols.

She also wants the feedback to be user-friendly, so she wants to include the actual password in the output. Help Amrita finish this password checker using Python's built-in string methods.

Character Types Considered:

Lowercase letters (a-z) Uppercase letters (A-Z) Digits (0-9) Special characters (from string.punctuation, e.g. @, !, #, \$)

Input Format

The input consists of a single string representing the user's password.

Output Format

The program prints the strength of the password in this format:

If the password length < 6 characters or fewer than 2 of the 4 character types, the output prints "<password> is Weak"

If password length ≥ 6 and at least 2 different character types, the output prints "<password> is Moderate"

If Password length ≥ 10 and all 4 character types present, the output prints "<password> is Strong"

Refer to the sample output for formatting specifications.

Sample Test Case

Input: password123

Output: password123 is Moderate

Answer

```
# You are using Python
import string
```

```
# Read input
password = input()
```

```
has_lower = any(char.islower() for char in password)
has_upper = any(char.isupper() for char in password)
has_digit = any(char.isdigit() for char in password)
has_special = any(char in string.punctuation for char in password)
```

```
types_count = sum([has_lower, has_upper, has_digit, has_special])
```

```
if len(password) >= 10 and types_count == 4:
    strength = "Strong"
elif len(password) >= 6 and types_count >= 2:
    strength = "Moderate"
else:
    strength = "Weak"
```

```
# Print result
print(f"{password} is {strength}")
```

Status : Correct

Marks : 10/10

4. 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

```
# You are using Python
```

```
# Read input and convert to list of integers
```

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

```
square = list(map(lambda x: x**2, numbers))
```

```
cube = list(map(lambda x: x**3, numbers))
```

```
# Print results
```

```
print(square, cube)
```

Status : Correct

Marks : 10/10