

Rajalakshmi Engineering College

Name: Mohamed Aaris
Email: 240701316@rajalakshmi.edu.in
Roll no: 240701316
Phone: 6383899392
Branch: REC
Department: I CSE FC
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 2_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. Problem Statement

Rohith is a data analyst who needs to categorize countries based on their population growth rates. Each country is assigned a unique code. Rohith will receive a code and corresponding data based on the code. If the data falls within specific thresholds, he needs to classify the country's priority level.

Your task is to write a program that reads a country code and its associated data, and then determines if the priority is "High" or "Low."

Thresholds: France: Priority is "High" if the percentage < 50, else "Low". Japan: Priority is "High" if life expectancy > 80, else "Low". Brazil: Priority is "High" if the urban population > 80, else "Low".

Input Format

The first line of input consists of an integer, representing the country code (1 for France, 2 for Japan, 3 for Brazil).

If the country code is 1,

- The second line consists of a floating-point value N, representing the percentage of the English-speaking population.

If the country code is 2,

- The second line consists of a floating-point value A, representing the average life expectancy in years.

If the country code is 3,

- The second line consists of a floating-point value P, representing the percentage of the urban population.

Output Format

The first line of output displays "Priority: High" or "Priority: Low" based on the input data.

If the country code is invalid, print "Invalid".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1

30.0

Output: Priority: High

Answer

```
a = int(input())
```

```
if a == 1:
```

```
    x = float(input())
```

```
    if x < 50:
```

```
        print("Priority: High")
```

```
    else:
```

```
        print("Priority: Low")
```

```
elif a == 2:
    y = float(input())
    if y > 80:
        print("Priority: High")
    else:
        print("Priority: Low")
elif a == 3:
    z = float(input())
    if z > 80:
        print("Priority: High")
    else:
        print("Priority: Low")
else:
    # Invalid country code
    print("Invalid")
```

Status : Correct

Marks : 10/10

2. Problem Statement

Alex is practicing programming and is curious about prime and non-prime digits. He wants to write a program that calculates the sum of the non-prime digits in a given integer using loops.

Help Alex to complete his task.

Example:

Input:

845

output:

12

Explanation:

Digits: 8 (non-prime), 4 (non-prime), 5 (prime)

The sum of Non-Prime Digits: $8 + 4 = 12$

Output: 12

Input Format

The input consists of a single integer X.

Output Format

The output prints an integer representing the sum of non-prime digits in X.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 845

Output: 12

Answer

```
a=input()
sum=0
for i in a:
    b=int(i)
    if b not in [2,3,5,7]:
        sum+=b
print(sum)
```

Status : Correct

Marks : 10/10

3. Problem Statement

Taylor is tasked with a mathematical challenge that requires finding the smallest positive number divisible by all integers from 1 to n.

Help Taylor to determine the smallest positive number that is divisible by all integers from 1 to n. Make sure to employ the break statement to ensure efficiency in the program.

Input Format

The input consists of a single integer, n.

Output Format

The output displays the smallest positive number that is divisible by all integers from 1 to n.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 10

Output: 2520

Answer

```
import math
def lcm(a,b):
    return (a*b)//math.gcd(a,b)
def small(n):
    a=1
    for i in range(2,n+1):
        a=lcm(a,i)
    return a
n=int(input())
print(small(n))
```

Status : Correct

Marks : 10/10

4. Problem Statement

Nisha is a mathematics enthusiast, eager to explore the realm of twin prime numbers. The objective is to develop a program that enables the discovery and presentation of twin prime pairs.

The program should take an integer 'n' as input and generate 'n' pairs of twin primes, displaying the pairs with a difference of 2 between them.

Input Format

The input consists of a single integer, n.

Output Format

The output displays the 'n' pairs of twin primes, the pairs with a difference of 2 between them.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 5

Output: 3 5

5 7

11 13

17 19

29 31

Answer

```
# Function to check if a number is prime
```

```
def is_prime(num):
```

```
    if num <= 1:
```

```
        return False
```

```
    for i in range(2, int(num ** 0.5) + 1):
```

```
        if num % i == 0:
```

```
            return False
```

```
    return True
```

```
# Function to find and print the first n twin prime pairs
```

```
def find_twin_primes(n):
```

```
    twin_primes = []
```

```
    num = 2 # Start checking from the first prime number
```

```
    while len(twin_primes) < n:
```

```
        if is_prime(num) and is_prime(num + 2):
```

```
            twin_primes.append((num, num + 2))
```

```
            num += 1 # Increment the number to check the next potential prime
```

```
# Print the twin primes
```

```
for pair in twin_primes:
```

```
    print(pair[0], pair[1])
```

```
# Input: number of twin prime pairs to find  
n = int(input())
```

```
# Output the twin primes  
find_twin_primes(n)
```

Status : Correct

Marks : 10/10