# Rajalakshmi Engineering College

Name: Padma Priya D

Email: 240701377@rajalakshmi.edu.in

Roll no: 240701377 Phone: 8668123104

Branch: REC

Department: I CSE FD

Batch: 2028

Degree: B.E - CSE



## NeoColab\_REC\_CS23221\_Python Programming

REC\_Python\_Week 4\_CY

Attempt : 1 Total Mark : 40 Marks Obtained : 40

Section 1: Coding

### 1. Problem Statement

Create a program for a mathematics competition where participants need to find the smallest positive divisor of a given integer n. Your program should efficiently determine this divisor using the min() function and display the result.

### **Input Format**

The input consists of a single positive integer n, representing the number for which the smallest positive divisor needs to be found.

## **Output Format**

The output prints the smallest positive divisor of the input integer in the format: "The smallest positive divisor of [n] is: [smallest divisor]".

Refer to the sample output for the exact format.

### Sample Test Case

Input: 24

Output: The smallest positive divisor of 24 is: 2

#### Answer

```
# You are using Python
n=int(input())
divisors=[]
for i in range(2,n+1):
    if n%i==0:
        divisors.append(i)
    if divisors:
        smallest_divisor=min(divisors)
print(f"The smallest positive divisor of {n} is: {smallest_divisor}")
```

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

#

```
# You are using Python
DOMESTIC_RATE=5.0
INTERNATIONAL_RATE=10.0
REMOTE_RATE=15.0
def calculate_shipping(weight,destination):
    if weight<=0:
        print("Invalid weight. Weight must be greater than 0.")
    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

#### 3. Problem Statement

Imagine you are tasked with developing a function for calculating the total cost of an item after applying a sales tax. The sales tax rate is equal to 0.08 and it is defined as a global variable.

The function should accept the cost of the item as a parameter, calculate the tax amount, and return the total cost.

Additionally, the program should display the item cost, sales tax rate, and total cost to the user.

Function Signature: total\_cost(item\_cost)

### **Input Format**

The input consists of a single line containing a positive floating-point number representing the cost of the item.

## **Output Format**

The output consists of three lines:

"Item Cost:" followed by the cost of the item formatted to two decimal places.

"Sales Tax Rate:" followed by the sales tax rate in percentage.

"Total Cost:" followed by the calculated total cost after applying the sales tax, formatted to two decimal places.

Refer to the sample output for formatting specifications.

## Sample Test Case

Input: 50.00

Output: Item Cost: \$50.00

Sales Tax Rate: 8.0% Total Cost: \$54.00

#### Answer

#

# You are using Python
SALES\_TAX\_RATE=0.08
def total\_cost(item\_cost):
 tax\_amount=item\_cost\*SALES\_TAX\_RATE
 total=item\_cost+tax\_amount
 return total
item\_cost=float(input())

```
total_cost = total_cost(item_cost)
print(f"Item Cost: ${item_cost:.2f}")
print(f"Sales Tax Rate: {SALES_TAX_RATE * 100}%")
print(f"Total Cost: ${total_cost:.2f}")
```

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

# You are using Python
s=input()
sub=input()
count=s.count(sub)
print(f"The substring'{sub}'appears{count} times in the text.")

Status: Correct Marks: 10/10