

IF100

For Loops - Practice #1

Introduction

The aim of this example is to practice sequences (i.e. list, string, range) and loops (for statements). The use of sequences and loops is due to the nature of the problem; that is, you cannot finish this practice assignment without using sequences and loops.

Description

In this exercise, you will write a Python program that will find the sum of multiples of two numbers between 0 and a number given by a user input. You will be inputting three integers: a stop number, and 2 multipliers. As an example, if the stop number is 10 and the multipliers are 2 and 5; then the natural numbers below 10 and multiples of 2 or 5 are 2, 4, 5, 6, and 8, and the sum of these multiples is 25.

Your program will start by asking the stop number. If this number is less than 1, then your program will display a message which states that the stop number cannot be less than 1. Otherwise, your program will ask for the multipliers and then, calculate the sum of multiples of them if the following conditions are satisfied:

- Multipliers cannot be less than 1.
- Multipliers cannot be equal to each other.

If these conditions are not satisfied, your program will show a proper message according to the error that the user did. Otherwise, after calculating the sum, the value of it will be displayed.

Please see "Sample Runs" section in order to understand the flow of the program, the inputs and outputs in a better way.

Inputs and Outputs

The inputs of your program and their order are explained below. Please see "Sample Runs" section for some examples.

There are three inputs to your program, in the order given below:

1. Stop number
2. 1st multiplier
3. 2nd multiplier

The prompts of the input statements to be used has to be exactly the same as the prompts of the "Sample Runs".

If the user enters a number less than 1, then your program should display an appropriate error message (specifically, *"Stop number can't be less than 1!"*) and exit. Please note that you cannot assume that all the inputs will be integer. If the user enters a non-integer value, then your program should display an appropriate error message (specifically, *"Stop number must be an integer!"*) and exit.

If the user provides an integer that is greater or equal to 1 as the stop number, there will be another input check for the multipliers. These checks will be done after getting both inputs from the user and the program will display an error message according to the multipliers. That is, the multipliers cannot be less than 1 and also they cannot be equal to each other. Additionally, please also note that you cannot assume that all the inputs will be integer. In such cases, the outputs will be as follows, respectively:

- *Multipliers cannot be less than 1!*
- *Multipliers cannot be equal to each other!*
- *Multipliers must be integers!*

If the multipliers are both less than 1 and equal to each other, then you should give the first error message (*"Multipliers cannot be less than 1!"*).

If the inputs satisfy all of these conditions, then your program will print the sum of multiples of 1st or 2nd multipliers. Hence, the output will be the following:

- *Sum of all multiples of 'multiplier1' or 'multiplier2' below 'stop_number' is 'result'.*

Sample Runs

Below, we provide some sample runs of the program that you will develop. The *italic* and **bold** phrases are inputs taken from the user. You may not change any of the prompt sentences. Your program should be presented exactly like these sample runs.

Sample runs are not %100 comprehensive. You are required to read the whole documentation and decide on what other cases you might try your program with.

Sample Run 1 (*invalid stop number*)

Please enter the stop number: **-8**
Stop number cannot be less than 1!

Sample Run 2 (*invalid stop number*)

Please enter the stop number: **a**
Stop number must be an integer!

Sample Run 3 (*invalid multiplier*)

Please enter the stop number: **12**
Please enter the first multiple: **-8**
Please enter the second multiple: **11**
Multipliers cannot be less than 1!

Sample Run 4 (*invalid multiplier*)

Please enter the stop number: **19**
Please enter the first multiple: **2**
Please enter the second multiple: **-4**
Multipliers cannot be less than 1!

Sample Run 5 (*invalid multiplier*)

Please enter the stop number: **13**
Please enter the first multiple: **5**
Please enter the second multiple: **5**
Multipliers cannot be equal to each other!

Sample Run 6 (*invalid multiplier*)

Please enter the stop number: **21**
Please enter the first multiple: **-2**
Please enter the second multiple: **-2**
Multipliers cannot be less than 1!

Sample Run 7 (*invalid multiplier*)

Please enter the stop number: **21**
Please enter the first multiple: **2**
Please enter the second multiple: **abc**
Multipliers must be integers!

Sample Run 8 (*valid inputs*)

Please enter the stop number: **7**

Please enter the first multiple: **9**

Please enter the second multiple: **10**

Sum of all multiples of 9 or 10 below 7 is 0.

Sample Run 9 (*valid inputs*)

Please enter the stop number: **18**

Please enter the first multiple: **3**

Please enter the second multiple: **9**

Sum of all multiples of 3 or 9 below 18 is 45.