Return

March 29, 2024

Simulation: Function with Return

Make a python program that will input two two integers and will create the following functions with the specified tasks.

a. Function that will compute the sum of two integers.

b. Function that will compute the product of two integers.

c. Function that will compute the difference if the first integer is greater than 2^{nd} integer.

d. Function that will compute the remainder if the 2^{nd} integer is greater than the first integer.

e. Function that will compute the quotient if the 1^{st} integer is greater than the 2^{nd} integer. No addition, multiplication, division and modulus operation with 0 value.

f. No arithmetic operation with 0 value.

[2]:

```python
from IPython.display import display, Math
```

[3]:

```python
# universal function for getting the input.
# this function is not included in the task.
# this to avoid redundancy in the code and make it more readable the ever before
def input_num():
    x = int(input("1st number: "))
    y = int(input("2nd number: "))
    return x, y
```

[4]:

```python
# for flagship, check if the value is 0
# this function is not included in the task.
def is_zero(x, y):
    if x == 0 or y == 0:
        display(Math(r'\text{No Operation Perform}'))
        return True
    return False
```

[5]:

```python
# program a
def twoSum():
    x, y = input_num()
    if not is_zero(x, y):
        return display(Math(str(x + y)))
```

```
# input value: x = 0, y = 5
twoSum()
```

No Operation Perform

[8]:

```
# program b
def twoProduct():
    x, y = input_num()
    if not is_zero(x, y):
        return display(Math(str(x*y)))
# input value: x = 33, y = 25
twoProduct()
```

825

[9]:

```
# program c
def twoDiff():
    x, y = input_num()
    if not is_zero(x, y):
        if x > y:
            return display(Math(str(x-y)))
        else:
            return display(Math(fr'\text{{{x} is not greater than {y}}} \text{{␣
 ↪value.}}'))
# input value: x = 33, y = 125
twoDiff()
```

33 is not greater than 125 value.

[10]:

```
# program d
def compute_remainder():
    x, y = input_num()
    if not is_zero(x, y):
        if y > x:
            return display(Math(str(y%x)))
        else:
            return display(Math(fr'\text{{{y} is not greater than {x}}} \text{{␣
 ↪value.}}'))
# input value: x = 23, y = 110
compute_remainder()
```

18

[11]:

```
def compute_quotient():
    x, y = input_num()
    if y == 0 or x == 0:
        return display(Math(r'\text{No arithmetic operation with } 0 \text{␣
 ↪value.}'))
    elif x > y:
        return display(Math(str(x / y)))
    else:
        return display(Math(fr'\text{{{x} is not greater than {y}}} \text{{␣
 ↪value.}}'))
# input value: x = 27, y = 10
compute_quotient()
```

2.7

[13]:

```
def compute_quotient():
    x, y = input_num()
    if y == 0 or x == 0:
        return display(Math(r'\text{No arithmetic operation with } 0 \text{␣
 ↪value.}'))
    elif x > y:
        return display(Math(str(x / y)))
    else:
        return display(Math(fr'\text{{{x} is not greater than {y}}} \text{{␣
 ↪value.}}'))
# input value: x = 13, y = 210
compute_quotient()
```

13 is not greater than 210 value.

Author: Willie M. Bonavente