

Yuguan Pan

2022-Mar-13

IT FDN 110 B Winter 2022: Foundations of Programming

Assignment07 Knowledge Document

## **Lesson learned in finishing assignment07**

### **Introduction**

Brief overview of some lessons learned through some experience I earned through finishing assignment07.

### **Experience gained 1:**

I modified the script using try-except function. This loop is useful when you are not sure whether the code will go through or not. If it works, then it will execute all the code in the try loop. Otherwise, it will intentionally interrupt the program and throw the exception information in the except loop to avoid returning error and crash of program.

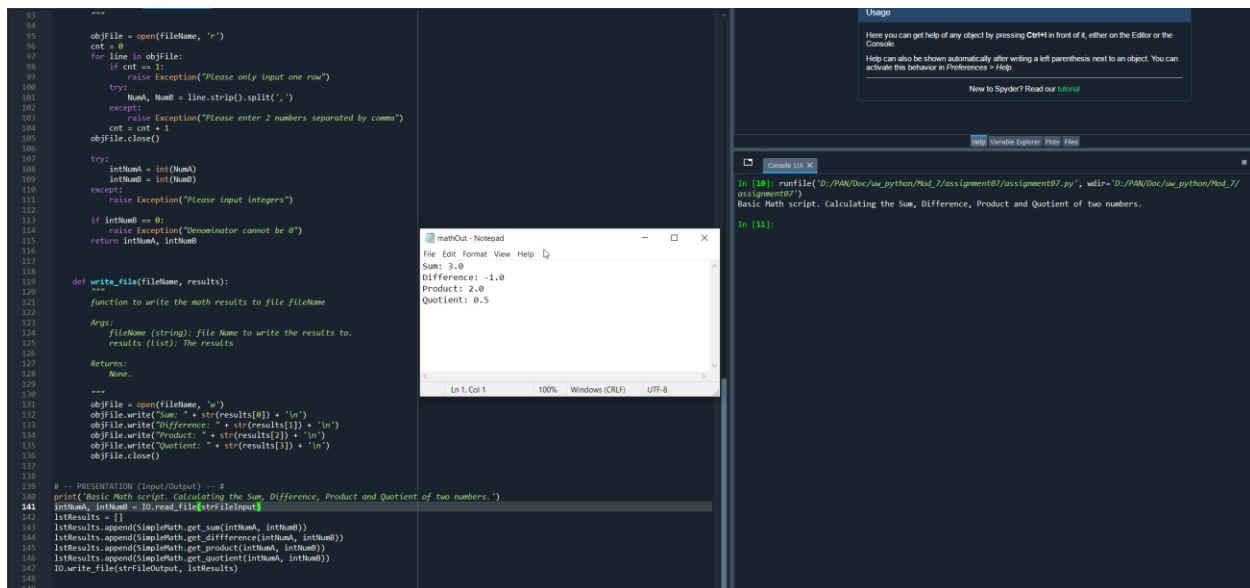
### **GitHub link:**

[yuguanp/assignment07 \(github.com\)](https://github.com/yuguanp/assignment07)

### **Summary**

When doing modifying based on someone else's work it is very easy to generate bugs since some of the original scripts may not fit the new ones I add in, and causing functional scripts become bugs. Thus instead of spending time on debug later, reading through and try to understand the original scripts before modifying could be a smarter way.

## Appendix



```
01  """
02
03  objFile = open(fileName, 'r')
04  cnt = 0
05  for line in objFile:
06      if cnt == 1:
07          raise Exception("Please only input one row")
08      try:
09          NumA, NumB = line.strip().split(',')
10      except:
11          raise Exception("Please enter 2 numbers separated by comma")
12      cnt = cnt + 1
13  objFile.close()
14
15  try:
16      intNumA = int(NumA)
17      intNumB = int(NumB)
18  except:
19      raise Exception("Please input integers")
20
21  if intNumA == 0:
22      raise Exception("Denominator cannot be 0")
23  return intNumA, intNumB
24
25 def write_file(fileName, results):
26     """
27     function to write the math results to file fileName
28
29     Args:
30         fileName (string): file name to write the results to.
31         results (list): The results
32
33     Returns:
34         None.
35
36     """
37     objFile = open(fileName, 'w')
38     objFile.write("Sum: " + str(results[0]) + "\n")
39     objFile.write("Difference: " + str(results[1]) + "\n")
40     objFile.write("Product: " + str(results[2]) + "\n")
41     objFile.write("Quotient: " + str(results[3]) + "\n")
42     objFile.close()
43
44 # -- PRESENTATION (Input/Output) -- #
45 print('Basic Math script: Calculating the Sum, Difference, Product and Quotient of two numbers.')
46 intNumA, intNumB = IO.read_file(strFileInput)
47 lstResults = []
48 lstResults.append(SimpleMath.get_sum(intNumA, intNumB))
49 lstResults.append(SimpleMath.get_difference(intNumA, intNumB))
50 lstResults.append(SimpleMath.get_product(intNumA, intNumB))
51 lstResults.append(SimpleMath.get_quotient(intNumA, intNumB))
52 IO.write_file(strFileOutput, lstResults)
53
54 """
```

mathOut - Notepad

```
File Edit Format View Help
Sum: 3.0
Difference: -1.0
Product: 2.0
Quotient: 0.5
Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

Usage

Here you can get help of any object by pressing **Ctrl+H** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in **Preferences > Help**.

New to Spyder? Read our [tutorial](#)

Console (1/4 X)

```
In [10]: runfile('D:/PAN/Doc/ae_python/Mod_7/assignment07/assignment07.py', wdir='D:/PAN/Doc/ae_python/Mod_7/assignment07')
Basic Math script: Calculating the Sum, Difference, Product and Quotient of two numbers.

In [11]:
```

Fig.1 Screenshot of script successfully running in Spyder

For terminal window, it automatically closed the prompt since there is nothing output in the terminal.