Kyle Biondich

7/18/2023

IT FDN 110 A

Assignment 02

Python Data Types

Introduction

Week 2 of the course introduced the different data types that exist in programming and how they are generally used when writing programs. The following paragraphs outline the methods that were used to transform between data types to successfully run each calculation.

Intended Outcome

The intended outcome for this week is a script file that presents the user with two prompts, each asking for a number. Then the script performs four mathematical operations, summing the numbers, differencing the number, multiplying the numbers, and lastly, dividing the numbers.

```
Enter a first number: 19
Enter second number: 33

The sum of 19 and 33 is: 52
The difference between 19 and 33 is: 14
The product of 19 times 33 is: 627
The quotient of 19 divided by 33 is: 0.5757575757575758
PS D:\DEV\uw\assignments\_PythonClass\Assignment02>
* History restored
```

Figure 1: Intended Outcome

Input Variables

After the header section, I begin by defining the variables that will capture the numbers that the user is asked to provide.

```
 9  ## Capture input from the user and convert the responses to either floats or integers
10  response1 = input("Enter a first number: ")
11  response2 = input("Enter second number: ")
12
```

Figure 2: Capturing Responses

Converting Strings

I had to google a number of ways that python converts strings into numbers. In my search, I was trying to find the simplest solution, which (Checking if a string can be converted to float in python, n.d.) eventually helped me out. Here python will try to convert the string to an integer, and if that fails, will then try to convert the string to a float. The variables 'number1' and 'number2' use the conversion to store the values as numbers for the rest of the script.

```
12
13  # This function first tries to convert the string to an integer and if that fails then tries to
14  # convert it to a float.
15  def convert_string(string):
16      try:
17          return int(string)
18      except ValueError:
19          return float(string)
20
21  # Provide converted response for the rest of the script
22  number1 = convert_string(response1)
23  number2 = convert_string(response2)
24
```

Figure 3: Trying to Convert Strings

Math Functions

The math functions are defined in Figure 4. I discovered the use of F strings and how they provide a simpler method of writing text and variables together.

```
25  # Begin defining each of the math steps as functions
26
27  # Compute the sum of the two inputs
28  def sum_variables (sum_var1, sum_var2):
29
30      sum_result = sum_var1 + sum_var2
31
32      ## Print the result
33      print(f'\nThe sum of {sum_var1} and {sum_var2} is: {sum_result}')
34
35  # Find the difference of the two inputs
36  def dif_variables (dif_var1, dif_var2):
37
38      # find the absolute value of the difference between the values
39      dif_result = abs(dif_var1 - dif_var2)
40
41      # print the result
42      print(f"The difference between {dif_var1} and {dif_var2} is: {dif_result}")
43
44  # Find the product of the two inputs
45  def prod_variables (prod_var1, prod_var2):
46
47      # find the product of value1 * value2
48      prod_result = (prod_var1 * prod_var2)
49
50      # Print the result
51      print(f"The product of {prod_var1} times {prod_var2} is: {prod_result}")
52
53  # Find the quotient of the two inputs
54  def quot_variables (quot_var1, quot_var2):
55
56      # find the quotient of value1 divided by value2
57      quot_result = float(quot_var1 / quot_var2)
58
59      #Print the result
60      print(f"The quotient of {quot_var1} divided by {quot_var2} is: {quot_result}")
61
```

Figure 4: Math Functions

Defining the Main Function

Lastly, after all the math functions were created, I defined the "Main" function that would call each math function upon initialization. This would pass the values of value1 and value2 to these functions.

I then call the main function, and pass in the values of number1 and number2, which have already been converted to numbers previously.

```
62  # Define the main function and execute each of the functions above
63  def main(value1, value2):
64      sum_variables(value1, value2)
65      dif_variables(value1, value2)
66      prod_variables(value1, value2)
67      quot_variables(value1, value2)
68
69  # Call the main function and pass in the converted response values
70  main(number1, number2)
```

Figure 5: Calling the Main Function

Summary

In summary, utilizing all the resources provided to the class and the online lecture, this paper outlines all the steps that were taken to create a python script that results in a successful execution of the intended outcome (Figure 1). Following the steps outlined above will allow for the audience to recreate the presented result.

References

Checking if a string can be converted to float in python. (n.d.). Retrieved from Stackoverflow: https://stackoverflow.com/questions/736043/checking-if-a-string-can-be-converted-to-float-in-python