New York University
School of Continuing and Professional Studies
Division of Programs in Information Technology

Introduction to Python

Homework Discussion, Session 1

This document provides an overview of each assignment and hints and tips to help you get started and work through problems.

Overall your job is to construct a logic machine using the features we have discussed this week. Mastery at this stage is simply a matter of becoming familiar enough with each feature's behavior that you can begin to use them as if you were using words in a sentence or tools from a toolbox. When it becomes available, you can refer to the Executive Summary for reference on each feature discussed in this course.

1.1 Exponientation and tidy border. So, a lot of steps here and you may not know where to begin. First, understand that this assignment is all about type conversion. Basically, a string can't be used in math expression (like the exponientation calculation done with **) but an integer can, and you can convert a string to **int** with **int()**. You can't get the length of a number with **len()** (which is needed if you're going to calculate the length of a "tidy border", but you can get the length of a string, and you can convert a number to a string with **str()**.

If you're not sure where to begin, you can try to follow this train of thought: start with the values you'll be getting from **input()**. What type of object is returned from **input()**? Go ahead and write a program that takes the two values required, and then verify the type of each of the values that are returned from **input()**. (Remember, you can check the type of any variable either by printing it directly, or by using **type()**. You can also check the Executive Summary (when it becomes available)**.** If you want to run a ** calculation with these values as operands, how must they be converted, and which function does that? Once the exponientation calculation is accomplished, what type of object comes back from **? If you want to create a custom-length border, you'll want to use the * operator to repeat the '**=**' character a certain number of times. The length of the border will be based on the length of the exponientation result, but you can't get a **len()** of an integer, so what type of value is required? How can the exponientation result be converted?

Finally, here is a "pseudocode" outline of the solution:

take user input value 1
take user input value 2

convert user input value 1 to an int
convert user input value 2 to an int

raise the 1st int to the power of the 2nd int (exponientation)

convert the exponientation result to a str
get the len of the str

repeat the border character ('=') by the len of the str, and print the result
print the exponientation value
print the custom border again (as above)

1.2 Tip calculator. This is a (relatively) simple calculation. Ignoring tax, the calculation is:

```
(bill + (bill * (tip_multiplier*.01))) / num_guests
```

However, keep in mind that you don't need to calculate the entire thing in one line using the formula above. You can (and probably should) break them out into separate calculations, assigning each result to a variable which you will then use in a further calculation.

The values for **bill**, **tip_multiplier** and **num_guests** are all input through **input()**. Since we're doing math calcuations with these values, they all have to be converted to numeric types (**int** and **float**).

So your program will take user input, convert to numeric types, convert the **tip multipler** value to a percentage multipler so it can be used in the formula above, and then report the result of the calculation.

Finally, here is a pseudocode outline of the solution:

take user input for value 1 (bill amount)
take user input for value 2 (number of people in your party)
take user input for value 3 (tip percentage, expressed as an integer)

convert the bill amount from string to a float
convert the number of people in your party to an int
convert the tip percentage to a float

multiply the tip percentage value by **0.01** so it can be used as a multipler (for example, if the user inputs **20** it should be converted to **.2** so it can be used to calculate the tip)

calculate the tip amount by multiplying the bill amount by the tip multiplier

sum up the total bill by adding the bill amount to the tip amount

divide the total bill by the number of guests to produce the "bill share".

Round the "bill share" to 2 decimal places to ensure that it's a dollars-and-cents value.

Use the **str()** function and the **+** concatenation operator to connect the numeric values with strings explaining the calculated result.