Andrew Collins

6-Dec-20

Foundations of Programming: Python

Assignment 06

**Introduction**

I found this to be the most challenging of the assignments yet, as evidenced by the late submission (again).

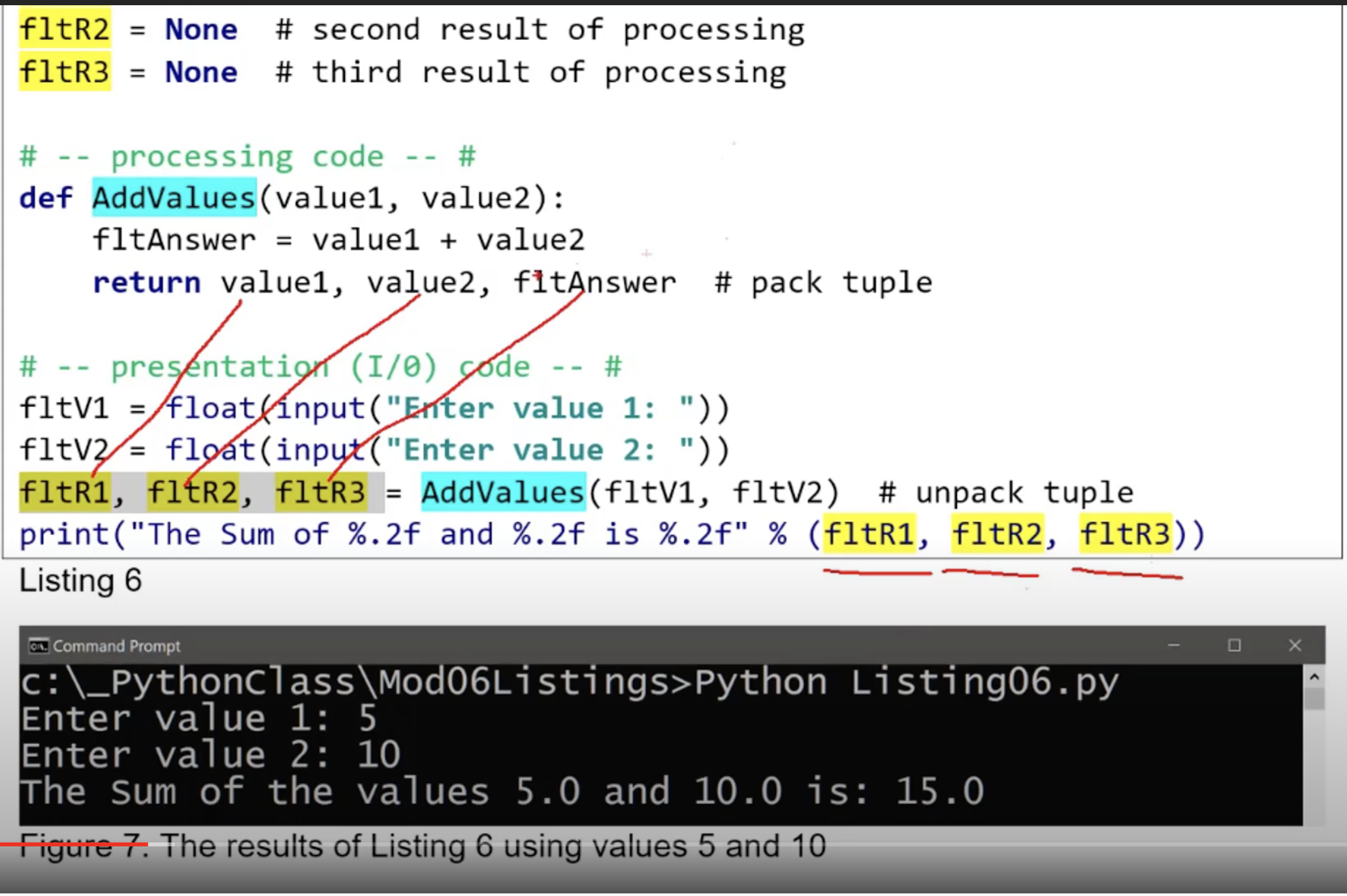
**Discussion and notes**

I found several confusing aspects of this module during the module video, which I took note of and was later able to answer. I’m attaching the notes at the end of this document as examples of what I was confused by and what I was able to understand.

After several failed attempts, I wound up resorting to the answer that was supplied on the course website. By copying and pasting and working through each step, I was able to understand the workflow better than by watching the review video. I gained a better understanding of the organizational layout of Data, Processing, Presentation, and Main Body of Script and why it is best to have code laid out this way.

I still haven’t fully worked out using the debugger function, but am sure with practice that will improve.

**Notes from Mod 06 video and practice**

* Parameter is defined as variable being used by the function as a parameter (?)
  + Description of the job the function is performing
* Arguments are used to make a fxn perform different actions or return different results
* Example of confusion – in listing 5 –
* print ("The sum of %.2f and %.2f" % (fltV1, fltV2))
  + how are we supposed to know to use %.2f ?!
    - apparently the 2 in %.2f returns only 2 decimal points
    - so you can put in 3, or just %f and you’ll get three decs for v1 and none for v2
    - if you used %.s you’d get a string
* 19:51 comments on separating processing and presentation code using return for fxns
* 23:07 comments on packing and unpacking tuples 
* 29:20 positional vs. named arguments
  + Confused on this
    - fltV1 = argument
    - def AddValues = (value1, value2) – here value 1 and 2 are parameters
    - you can also call by naming via AddValues(value1 = fltV1)
      * this allows for order of parameters and arguments to no longer be important…the names are
    - otherwise the positional argument would be AddValues (fltV1, fltV2)
* None is a special data type used to indicate the absence of a parameter value
* 46:00 is a review of reference vs. data types (listing 12)
* **Global vs. Local variables**
  + Global variables pertain to the whole script (such as the ones defined at the top of the script), whereas local variables are constrained to being within defined functions. Outside of the function definition, if you try to use a local variable, it will return an error that the particular local variable was not defined.
* **Classes 1:06**

**Conclusion**

I’m still quite behind in this course, but feel that I mostly understand the concepts when they’re laid out – but struggle with being able to conjure them from memory. I think that having the road map laid out ahead of me, I’ll be able to revisit these lessons in the future and apply them to my work.