Adrian Ronsse

August 17, 2021

Foundations Of Programming: Python

Assignment 06

https://adrianronsse.github.io/ITFdn110-Mod06/

**Defining and Using Functions**

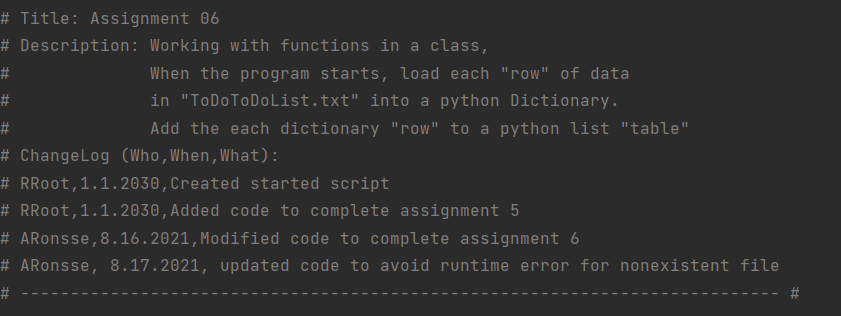
**Introduction:**

This paper details the steps to update an existing Python script with new code to utilize Python’s ability to define functions and adhere to “Separations of Concerns” best practices.

**Writing the code:**

Step 1 – Update Change Log

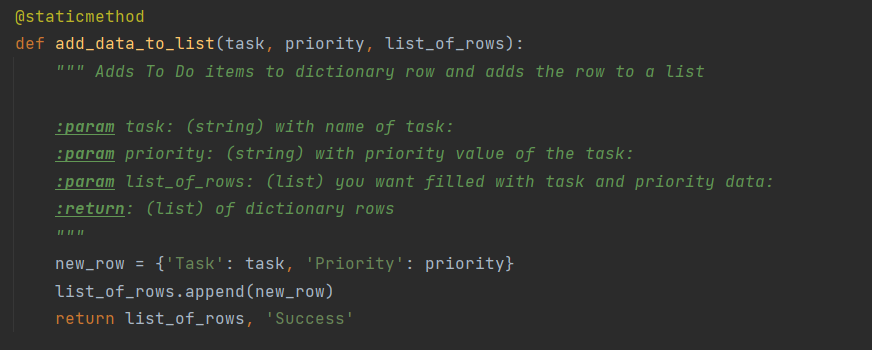
When changing anything in the file, I updated the change log to reflect my interactions with the file.



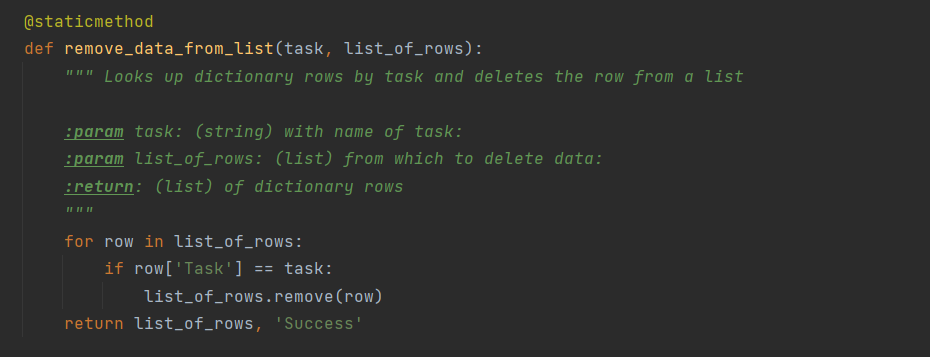
*Figure 1 – Update change log*

Step 2—Insert code in the “Processor” class functions where indicated

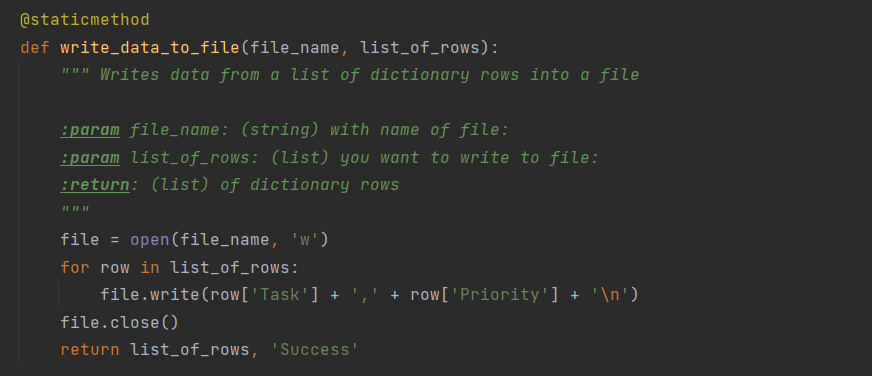
The first directive from Mr. Root’s pseudo code was to insert code into various functions he had already defined under the “Processor” class. I had already written the code in Assignment 5 to perform the processing tasks indicated, so I copied and pasted from my Assignment 5 code document, updating variable names so that they would not overlap with any of the global variables defined at the top of the Assignment 6 starter code provided by Mr. Root. I also documented each function using Mr. Root’s format for the read\_data\_from\_file function as a model for what data to represent in the docstring.



*Figure 2 – Processor function to add data*



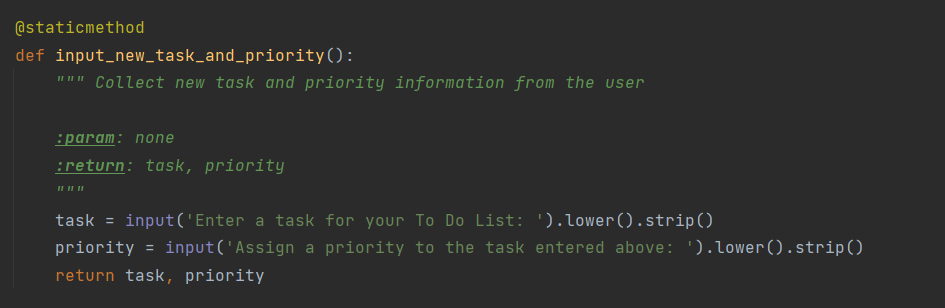
*Figure 3 – Processor function to remove data from list*



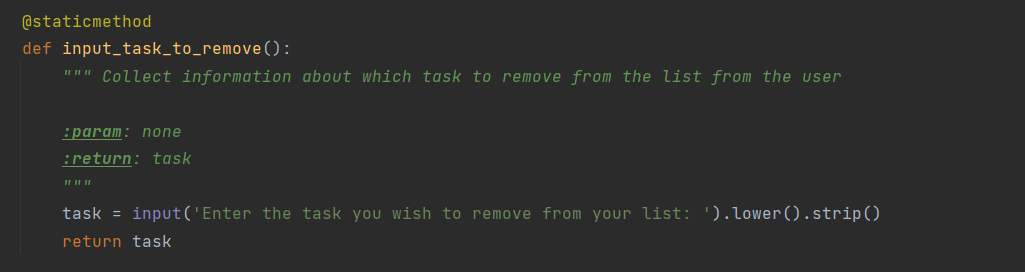
*Figure 4 – Processor function to save data to file*

Step 3—Insert code in the “IO” class functions where indicated

The next class of functions defined in the starter code prepared by Mr. Root was the “IO” class. This grouping of functions was intended to capture inputs from the user. Again, I had already written the core elements of this code in Assignment 5, so I copied the portion of the code requesting inputs from the user into the appropriate functions here, replacing any global variables listed at the top of the code, and including docstrings to describe each function. I did reuse local variables within the different functions but I think this is not too confusing since they are self-contained by nature.



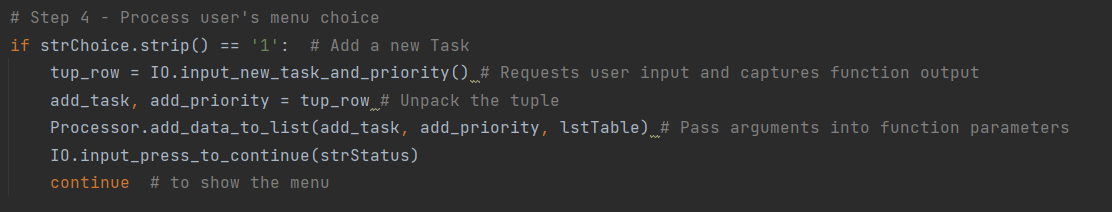
*Figure 5 – Collecting task and priority data*



*Figure 6 – Collecting request to remove data from list*

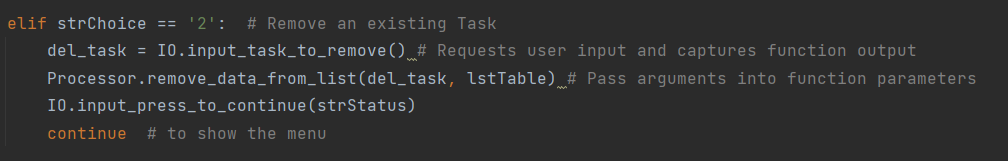
Step 3—Calling functions in the main body of the script

Finally it was time to use my functions. The first directive was for adding a task, so I called the IO function first to collect data from the user for processing. In so doing, I needed to capture the output as well. The output was produced as a tuple, so I then needed to unpack the tuple for processing. Finally I could call the processor function and pass the arguments from the tuple into the parameters for that function. I used new global variables to unpack the tuple, so I updated the declared variables at the top of the document to display all of the new variables I used in this portion of the code.



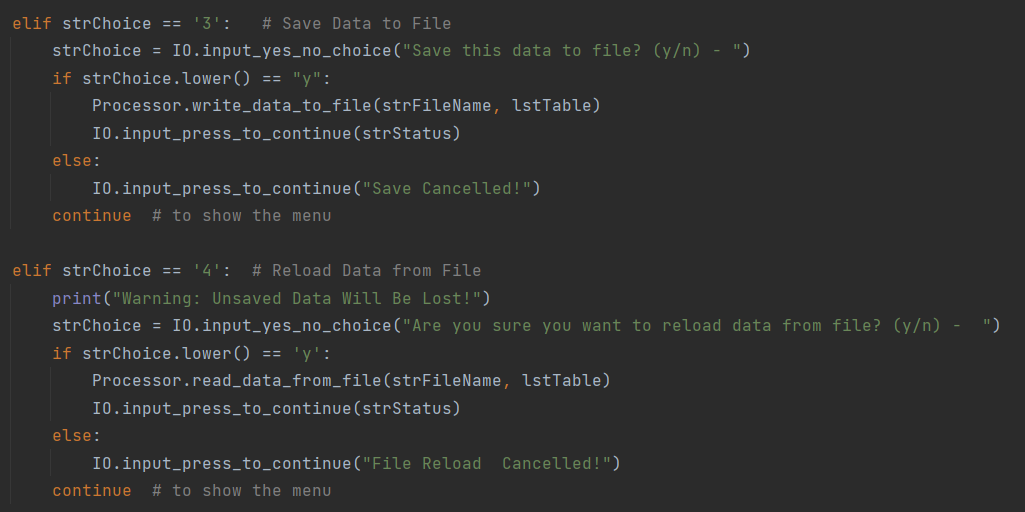
*Figure 7 – Capturing function output and passing it into the next function for processing*

The next action item was simpler, requiring only to collect a single piece of information from the user. I captured this using a new global variable del\_task so I also added this variable to the list at the top of the file.

**

*Figure 8 – Capturing function output and passing it into the next function for processing*

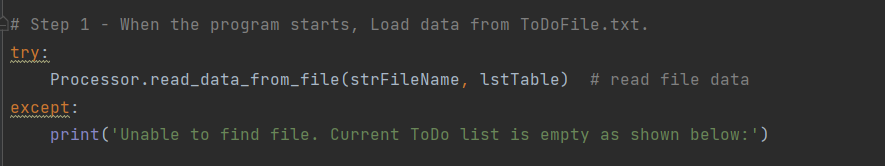
The final two action items were the most simple, requiring only to call the processing function and pass in the arguments corresponding to the file name and list built.

**

*Figure 9 – Using processing functions to write data to file and load data from file*

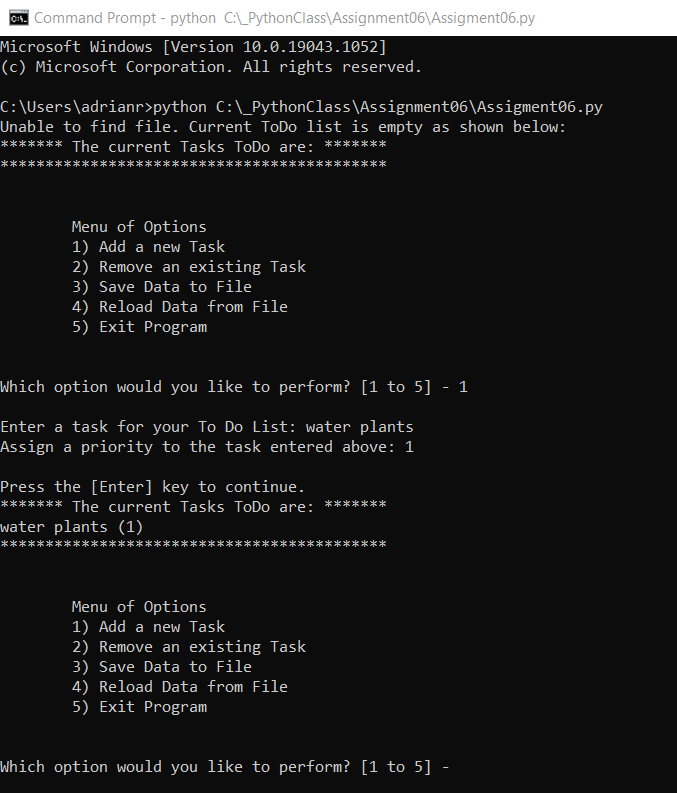
**Testing the code:**

After preparing the code, I needed to test it to make sure it was working as expected. First I received a runtime error because the file name defined in Randal’s code did not yet exist on my computer. To avoid this problem for future users, I updated that portion of the code with a try/except statement.



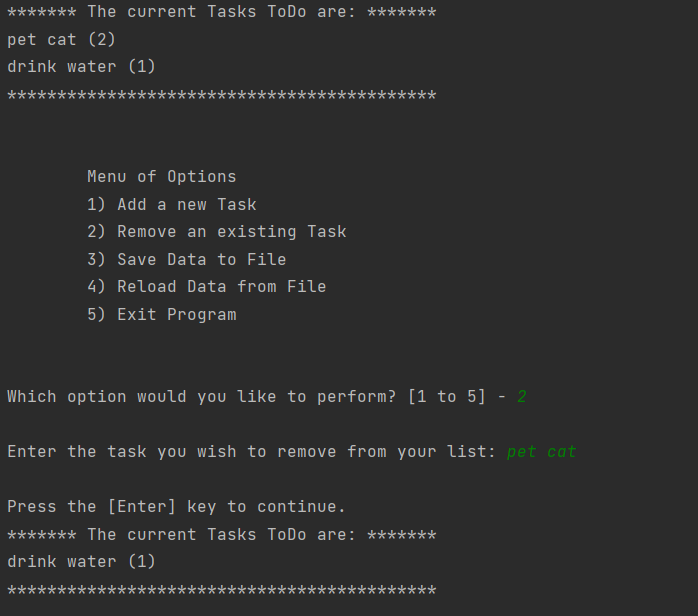
*Figure 10 – Using a try/except statement to avoid runtime error for missing file*

After updating the code as shown, I was then able to run the code in the command console as shown below:



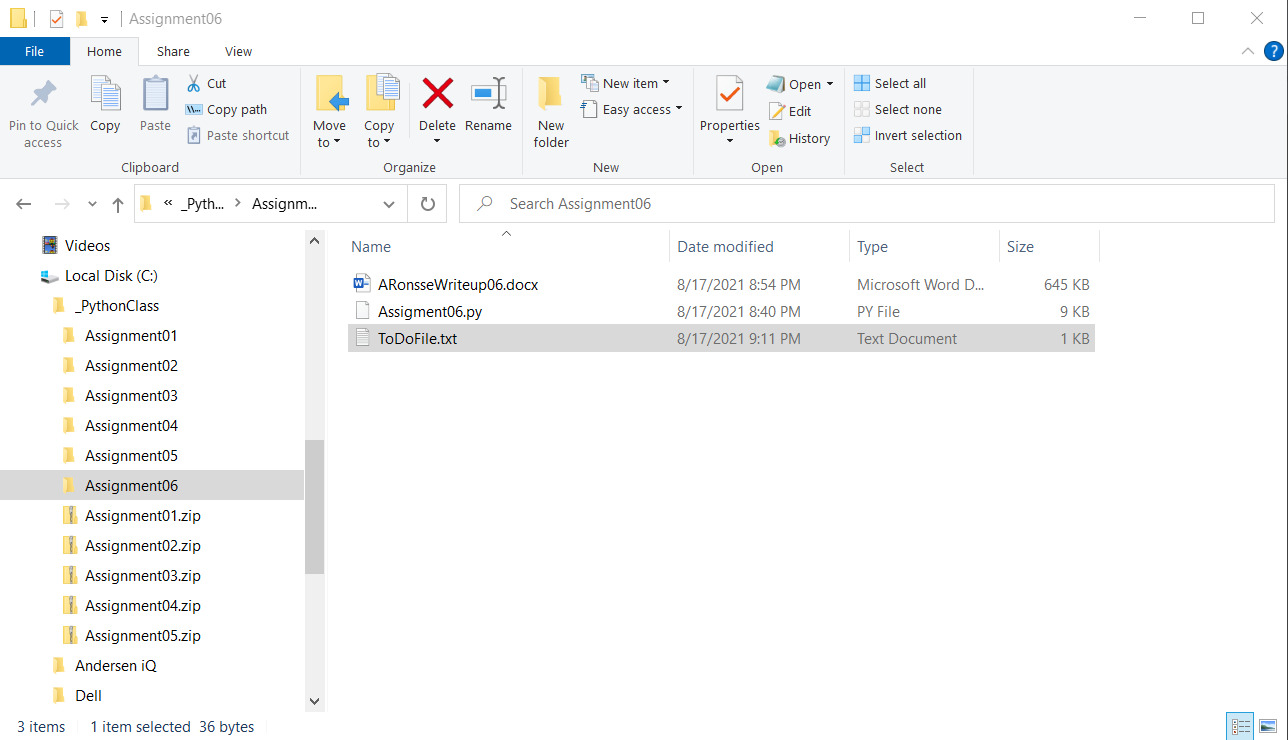
*Figure 11 – Command console test*

Then I tested it in PyCharm as well and found I was able to perform all of the menu items (deleting item shown below):



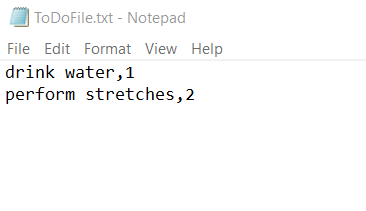
*Figure 12 – PyCharm test*

Finally I needed to check that the file had been created and that it stored the data as intended. Here I see that the .txt file has been created in my assignment folder:



*Figure 13 – Checking new file location*

Then I opened the file to make sure the data was contained:



*Figure 11 – File contains data*

**Summary:**

In this assignment I was able to update a program to define functions for processing data and for collecting input from the user, then use those functions to perform tasks in the main body of the code.