Sheridan Day
May 24, 2022
Foundations of Programming
Assignment 06
GitHub: https://github.com/SheridanDay/IntroToProg-Python

Document Your Knowledge

Introduction

Week six intimidated me due to my issues with week five. I was worried I'd be stuck again, so I put it off a few extra days. Bad idea, maybe. However, once I got in I was able to add my code relatively quickly compared to my days of suffering the week before. However there are always the little things to catch me up.

Topic 1 – More of the Same

Right off the bat I realized things were basically doing last week's assignment, but with more flair. I popped open my Assignment05 code and went to work piecing together what I would need to add into the processes for this week. There were many updates, such as changing my strTask to task. I chose to keep some elements that helped me remember what was what, such as dicRow (Figure 1).

```
    :param file_name: (string) with name of file:
    :param list_of_rows: (list) you want filled with file data:
    :return: (list) of dictionary rows
    """
    # TODO: Add Code Here!
    objFile = open(file_name, "w")
    for dicRow in table_lst:
        objFile.write(dicRow["Task"] + "," + dicRow["Priority"] + "\n")
    return list_of_rows
```

Figure 1. I kept dicRow in my script.. It didn't affect the final outcome and helped remind me this was a dictionary.

Topic 2 – I Don't Like Snake Casing

Oh snake casing. I do not like having so many weird looking gaps in my code. I kept it all from the original Starter code so as not to break anything and force myself to start over. In the grand scheme of

things, it's not the end of the world. It's more of a looks and presentation. Additionally, I kept screwing up names. Specifically. I kept flipping the table name and had to chase down my own errors while running the code. It was funny the first time. It was not funny the seventh time. I did learn eventually, or at least I ran out of errors to fix (Figure 2).

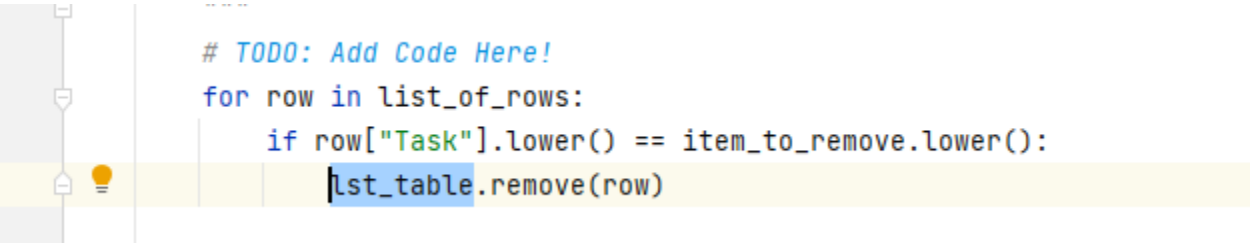

Figure 2. No it's not lst_table, it's table_lst.

Topic 3 – Copy & Paste is Great Until It's Not

As stated above, I copied/pasted pieces of working code from Assignment05 into this assignment. I also scrolled down through the provided info and prepped the names I would need for specific objects to reference above. I hoped I could catch those naming conventions before I ran things (Figure 3).

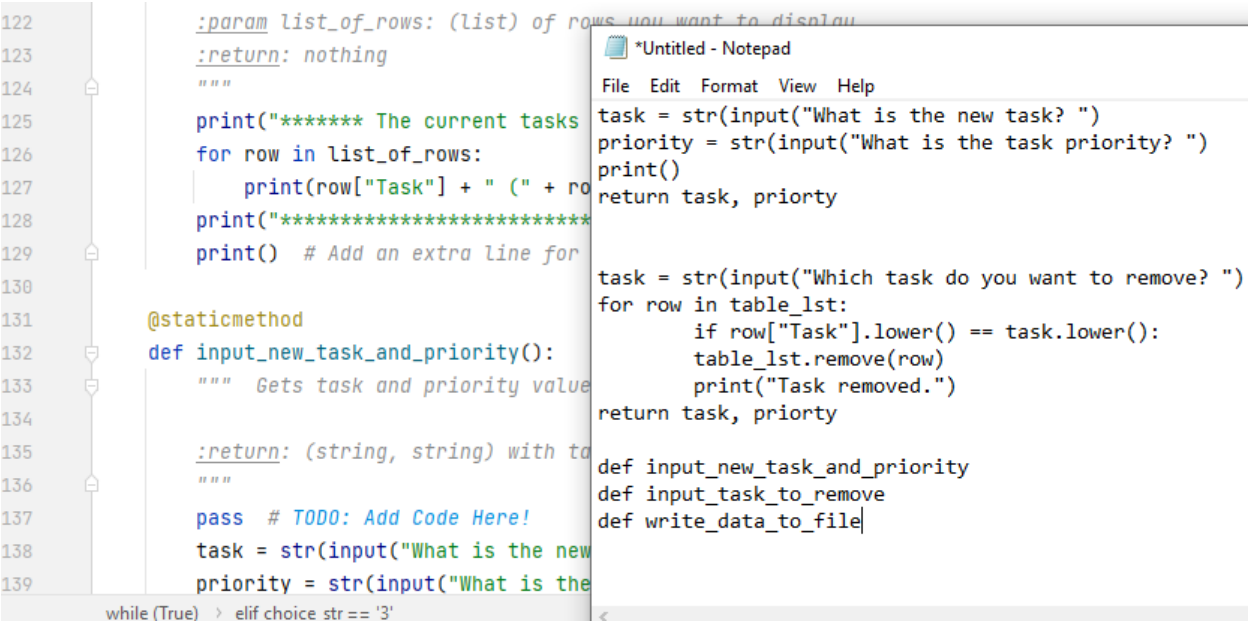

Figure 3. My PyCharm on the left and Notepad on the right organizing my code before pasting.

Of course, not everything was caught before my test runs. I missed a couple of strTask references. There were some missing brackets to close inputs. I referenced a Processor issue instead of the IO issue. And, of course, the classic table caught up to me a few times (Figure 4).

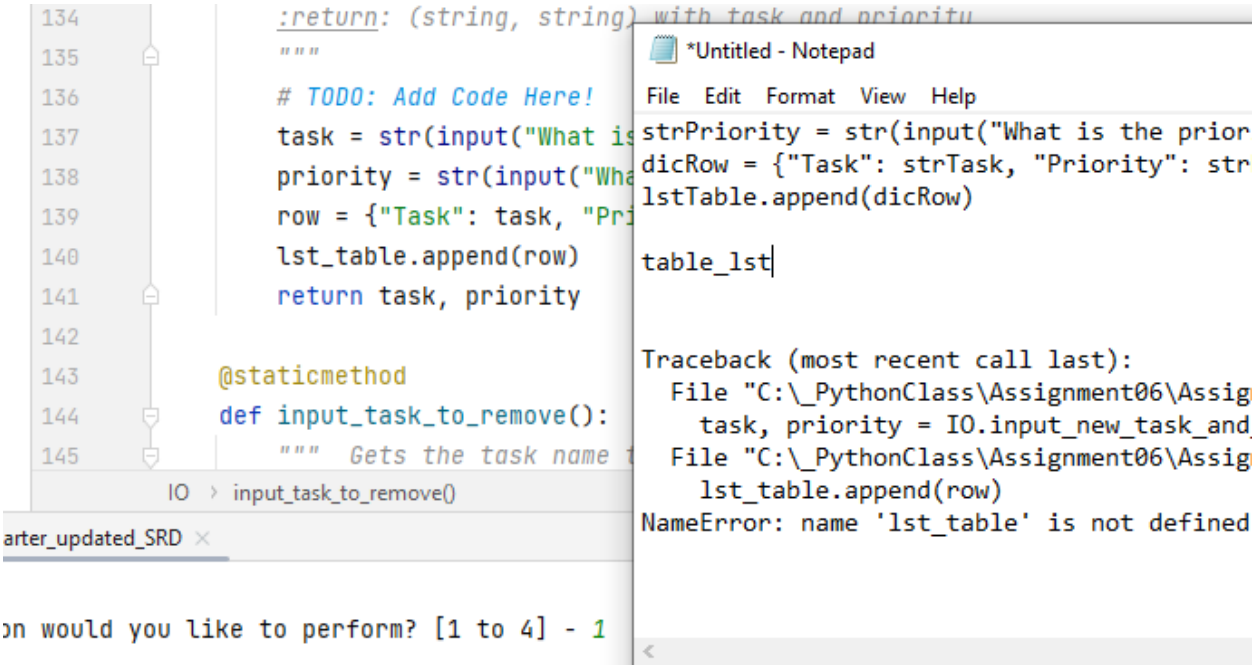

Figure 4. Error from PyCharm is in Notepad so I can scroll up and see my bad references again. Yes I added table_lst so I could copy/paste when I found another error. Above on Notepad, older code with the old lstTable that may have also confused me.

Topic 4 – I Don't Mind Using Others' Code

Last week was a struggle because a lot of the code was provided and we had to fill in the gaps. It was like doing a crossword in a different language when you only know the basic vocab. This week I could follow each step and understand what the main body of the script wanted so I could tell it to do those things in the process steps. Even it was helpful, I'm glad we didn't have to do all those pieces just yet. I would for sure be using a Notepad with all my names and references, like Figure 3 above.

Summary

Overall, this week made more sense than I expected. Grouping processes together to run in their own chunks is a much cleaner looking way of creating scripts. I don't like the look of snake casing, but I at least could read through everything and ensure I was referencing everything as much as I needed. The code worked well ((Figure 5) and was another interesting lesson.

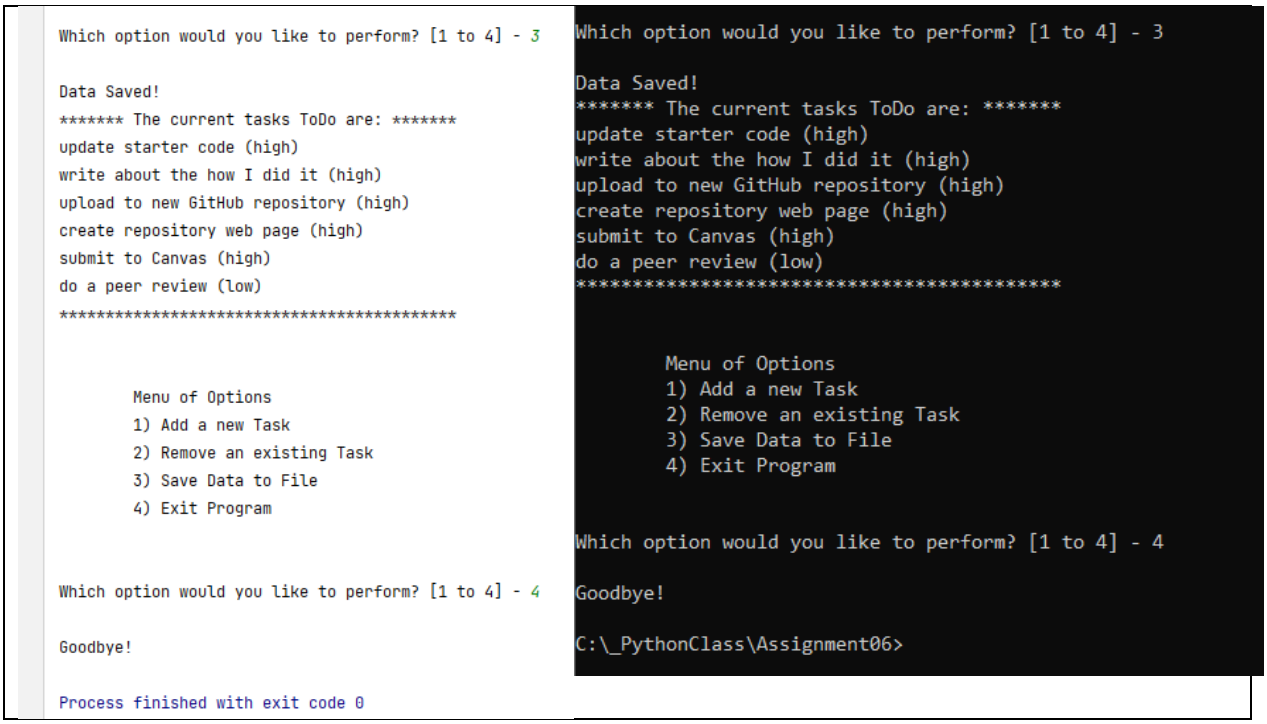

Figure 5. The code running in both PyCharm (left) and Command Prompt (right).