Hien Ho

August 8, 2022

Foundations of Programming: Python

Assignment 05

https://github.com/HHoUW/IntroToProg-Python

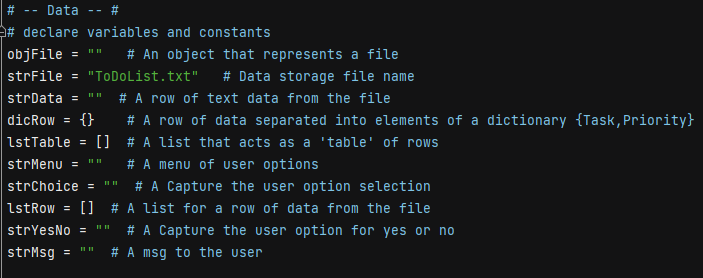
**To Do List Script**

**Introduction**

In this report, I will go over how I created a “To-Do List” program in Python. The program runs and presents a menu of options to the user. The options allow the user to review current tasks and their priority that are on the list, to add new tasks to the list, to remove tasks from the list, to save the list to a data file and to exit the program.

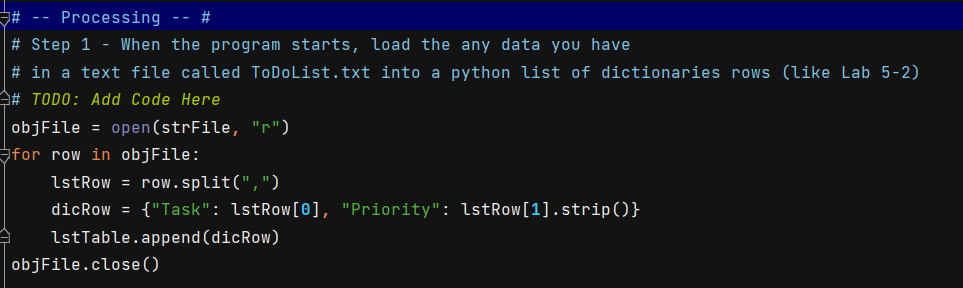
**Writing the code**

At the beginning of the script, the variables that will be use in the code are declared. (Figure 1)



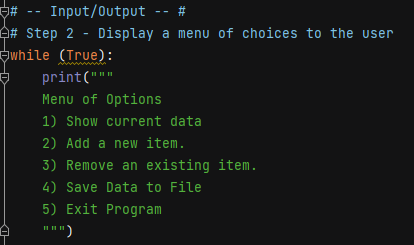
*Figure 1: Declaring variables.*

Next step is to open a data file called “ToDoList.txt” and read in the data that is in the file. This is accomplished with a **for loop**. The program will loop through reading one row of data from the data file at a time, split that string of data by the comma (“,”) in the string and store it in a List. The Dictionary gets populated with the data in the List. The keys use in the dictionary is “Task” and “Priority”. The Dictionary is added to another List using the **append()** method. This List acts as a table of rows of data. Once the loop is done and all data from the file have been read in, the program closes the data file. (Figure 2)



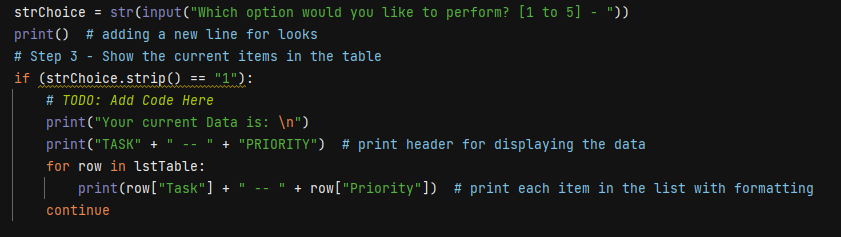
*Figure 2: Reading in data from a data file.*

The main body of the code will run in a **while loop**. This keeps the program running until the user chooses to exit the program. A menu of options is display to the user and the program waits for the user to make select an option. (Figure 3)



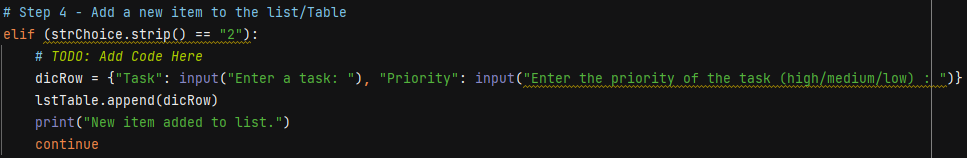
*Figure 3: Menu of options.*

Option 1 displays the current list of tasks to the user. A heading “TASK – PRIORITY” is printed and then a for loop is use to unpack the data from the List that contains the data. The Keys “Task” and “Priority” are used to extract the data from the List. Formatting is applied to the data and then display to the user. The program then returns the user to the main menu. (Figure 4)



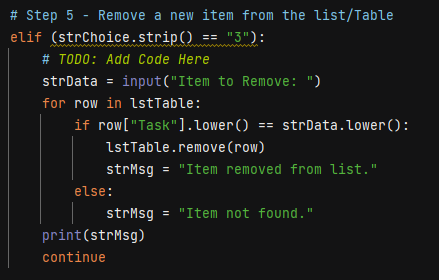
*Figure 4: Option 1, display the current list of data to the user.*

Option 2 allows the user to add a new item to the list of tasks. The user is prompted to enter a task and its priority. The user inputs get stored in a Dictionary. The Dictionary is then appended to the List that holds all the current data. Then the user is return to the main menu. (Figure 5).



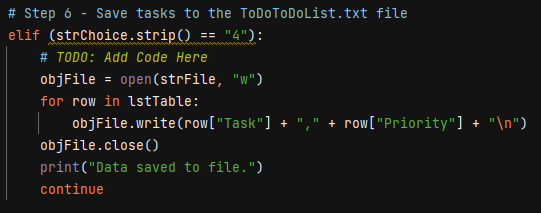
*Figure 5: Option 2, get new task from the user and add it to the list.*

Option 3 allows the user to remove an item from the list of tasks. User is prompted to enter a task to be removed. A for loop is use to compare the user’s input with each item in the List. If an item in the List matches the user’s input, that item is removed from the List using the **remove() method**. When the entire List have been loop through, the user gets a confirmation message if that item has been removed from the List or gets an item not found message when no matches were found. The user is then return to the main menu. (Figure 6)



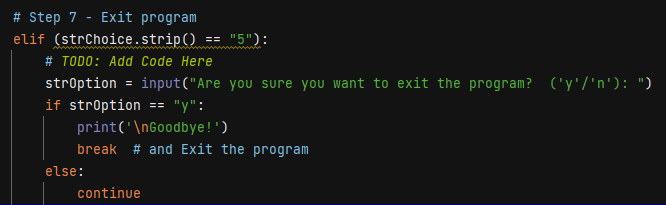
*Figure 6: Option 3, remove an item from the list.*

Option 4 saves the data that is currently in the List to a data file. The program opens the “ToDoList.txt” file again. The data file is overwritten with what is currently stored in the List. A for loop cycles through the List an item at a time and writes it as a new row in the data file. Once done, the data file is closed and the user gets a message saying that the data has been save to a file. The user is return to the main menu. (Figure 7)



*Figure 7: Option 4, save current data to a data file.*

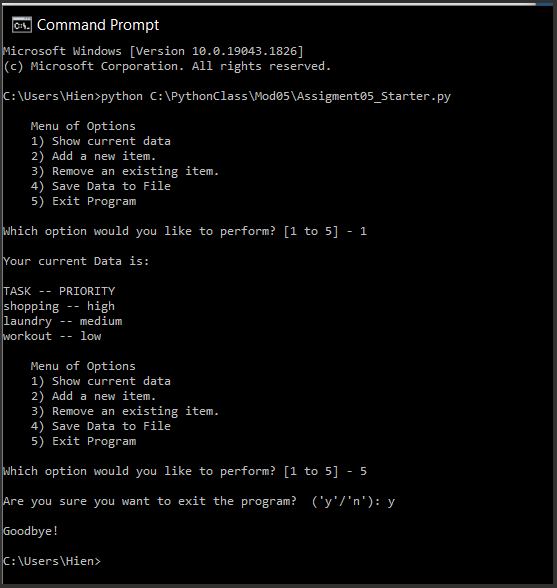
Option 5 ends and exits the program. The user is prompt with a message asking them if they are sure that they want to exit the program and to input either a “y” or “n”. If the user enters “y”, the user gets a goodbye message and the program ends. If “n” is entered, the program returns the user to the main menu. (Figure 8)



*Figure 8: Option 5, exit the program.*

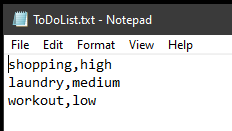
**Running the script**

I ran the script in both PyCharm and the command window and it ran properly on both. Here is the program running through all three options in the command window. (Figure 9)



*Figure 9: Program running in command window*

This is a screenshot of the text file to verify that data is successfully written to a text file. (Figure 10)



*Figure 10: Verifying the data is in the text file*

**Summary**

Using the knowledge that I have gain so far for this course, I was able to create a Python program to let a user to view and edit a to-do list. A list of tasks from a data file is read into the program and store using Dictionaries and Lists. The program gives the user options to review, edit and save the list of tasks back to the data file.