Neil McCracken

May 7, 2019

IT FDN 100 A Sp 19

Assignment06

GitHub: <https://github.com/mccrackenn/IntroToProg-Python/blob/master/ToDo.py>

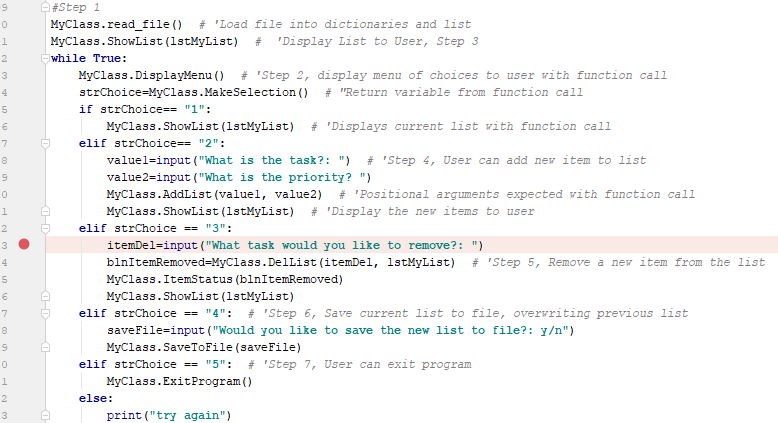
Organizing Data with Functions in a Class

# Introduction

Assignment 6 had us organize Assignment 5 through the use of functions grouped into a class. Inside of a class, functions are called methods. Functions/methods offer software reuse, which means that a programmer can repeatedly call upon the function without having to reinvent the wheel. For example I called upon my showList function four times through my program. That saves the writing of more code and complexity.

## While Loop Guide

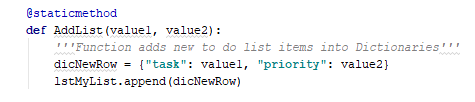
The challenge of this assignment, for me, was figuring out what to put into methods of MyClass and what not to put into the class. In addition much of the code is recycled from assignment five. Originally I tried to place everything into MyClass but I struggled with the while loop that guides the program. I eventually separated the while loop which acts as a traffic control for the program. Figure 1 below shows the while loop. The data from the text file is loaded into the program through the MyClass.read\_file call. The next call to MyClass.ShowList displays the current contents of the do list. I used this method four times in the body of the loop, employing it whenever I wanted to show the user changes made or for the option to display current data.



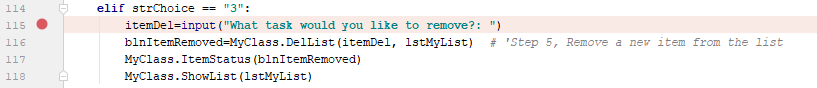
*Figure 1: Screenshot of while loop that guides the program*

### Static Methods

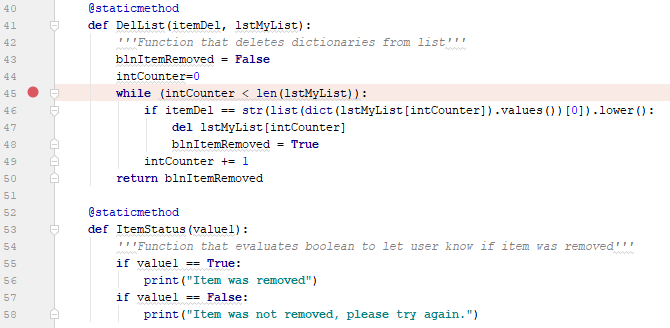
Static methods are used to process on the original as opposed to creating new copies/objects with the instance method (Python Video Module 6-14). A bit of reading ahead from the text *Python Programming for the Absolute Beginner,* “like all static methods, it’s designed to be invoked through a class and not an object.”(Chapter 8, pg. 231) All of the methods in my program are called with MyClass.*somecode().* Quite a few of the methods in my program use positional parameters in the header and position arguments when calling. I’m not going to screenshot and display every method in my program in this document as it would be redundant however I will cover a couple. In the call to AddList, which is used for adding new items to the list, I pass two variables. The two variables (value1and value2) pass the data that goes into the new dictionary item which gets appended to lstMyList (Shown in figure 2)



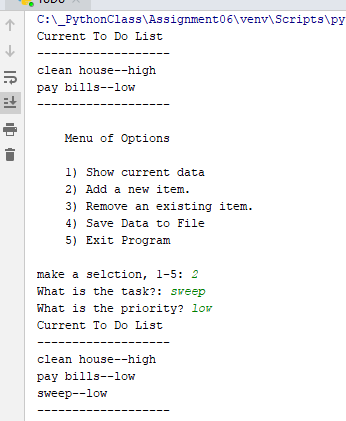
*Figure 2: Screenshot of AddList*

To delete items off the list I scrapped my script that I used from assignment five and used the provided template. I could not get my original to reliably delete items. The original is commented out. In the call on line 115 (Seen in figure 3), I want to catch the return value and use it on line 116 to let the user know if the item was deleted in the next call to MyClass.Items(blnItemRemoved). This method evaluates a Boolean expression and informs the user if the item was removed or not (Seen in Figure 4)

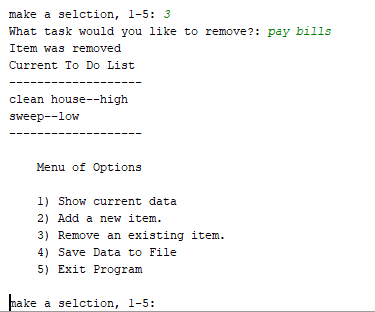
*Figure 3: Screenshot of calls to methods to delete an item and let user know the status.*

**

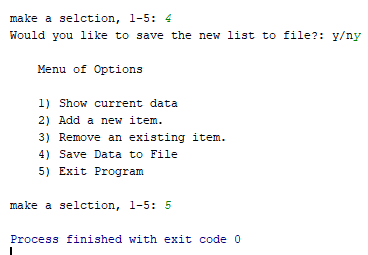
*Figure 4: Screenshot of DelList and ItemStatus*

**

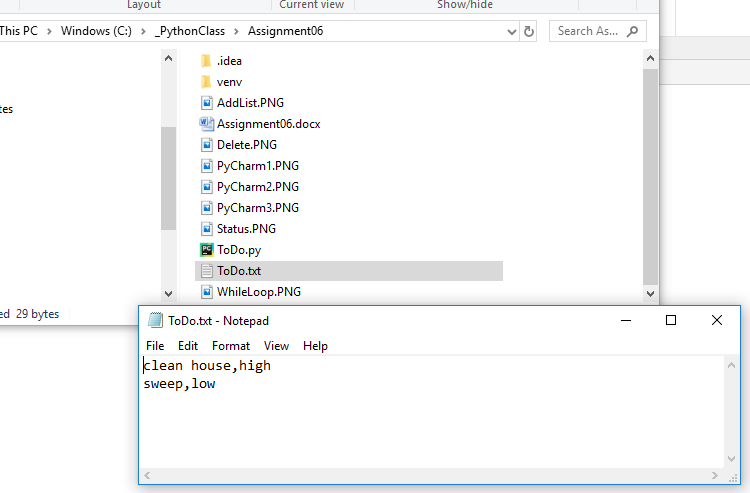
*Figure 5: Screenshot of Program starting in PyCharm*

**

*Figure 6: Screenshot of removing an item from the to do list*

**

*Figure 7: Screenshot of saving data and exiting program*

**

*Figure 8: Screenshot of updated list saved in correct location*

#### Conclusion

Assignment Six showed how functions/methods/classes provide an organizational system that allows data to be logically grouped. Code can be reused in the program along with other programs to save time. By using functions and methods we are meeting the goal of abstraction. The text book describes abstraction as giving us the ability to think about the big picture without having to worry about the small details. Only the call has to be written once the function is written.