Cameron Mathews August 17, 2021 IT FDN 110 A Su 21 Assignment06

https://github.com/cmathews10/IntroToProg-Python-Mod06.git

Modifying an Existing To-Do List Script and Adding Functions

Introduction

The following document outlines the steps taken to modify and update a Python script written by a different author with additional functions that will allow a user to select from a menu of options to input to-do list items and their associated priority level, edit the list, and store the input data into a text file.

Process

- 1. Write the script in PyCharm (Appendix A).
 - a. I found it more difficult to find the right locations to edit the formatting based on someone else's code.
 - b. It was difficult to determine if it was worth spending time to edit parts of the code that would have no functional output difference based on personal preference.
 - c. This assignment felt easier after working through corrections to the prior assignment, though I still feel confused on when to use tables, lists, dictionaries, etc. and need to spend more time researching these items.
- 2. Run the script in PyCharm (Appendix B).
- 3. Run the script in the Command Prompt window (Figure 3).

```
Microsoft Windows [Version 10.0.19042.1165]
(c) Microsoft Corporation. All rights reserved.
C:\Users\camer>python "C:\Users\camer\_PythonClass\Assignment06\main.py"
****** The current Tasks ToDo are: ******
Laundry (Mid)
Groceries (High)
Walk Dog (High)
************
       Menu of Options
       1) Add a new Task
       2) Remove an existing Task
       3) Save Data to File
       4) Reload Data from File
       5) Exit Program
Which option would you like to perform? [1 to 5] - 1
Task name: Mow Lawn
Task priority: Low
Success - the task has been added.
Press the [Enter] key to continue.
****** The current Tasks ToDo are: ******
Laundry (Mid)
Groceries (High)
Walk Dog (High)
Mow Lawn (Low)
*************
       Menu of Options
       1) Add a new Task
       2) Remove an existing Task
       3) Save Data to File
       4) Reload Data from File
       5) Exit Program
Which option would you like to perform? [1 to 5] - 2
What task would you like to remove? Laundry
Success - the task has been removed.
Press the [Enter] key to continue.
****** The current Tasks ToDo are: ******
```

```
Press the [Enter] key to continue.
****** The current Tasks ToDo are: ******
Groceries (High)
Walk Dog (High)
Mow Lawn (Low)
 ************
       Menu of Options
       1) Add a new Task
      2) Remove an existing Task
      3) Save Data to File
      4) Reload Data from File
       5) Exit Program
Which option would you like to perform? [1 to 5] - 3
Save this data to file? (y/n) - y
Success - the data has been saved.
Press the [Enter] key to continue.
****** The current Tasks ToDo are: ******
Groceries (High)
Walk Dog (High)
Mow Lawn (Low)
***************
       Menu of Options
       1) Add a new Task
       2) Remove an existing Task
       3) Save Data to File
       4) Reload Data from File
       5) Exit Program
Which option would you like to perform? [1 to 5] - 4
Warning: Unsaved Data Will Be Lost!
Are you sure you want to reload data from file? (y/n) - y
Success.
Press the [Enter] key to continue.
****** The current Tasks ToDo are: ******
Groceries (High)
Walk Dog (High)
Mow Lawn (Low)
************
```

```
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Reload Data from File
5) Exit Program

Which option would you like to perform? [1 to 5] - 5

Goodbye!

C:\Users\camer>
```

Figure 3: Example of the script successfully running in the Command Prompt window.

4. Confirm user input is being saved to the defined file location (Figure 4).

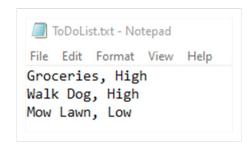


Figure 4: Image of the text file that updates based on user input.

Summary

This document outlined the steps taken to modify and update a Python script written by a different author with additional functions that will allow a user to select from a menu of options to input to-do list items and their associated priority level, edit the list, and store the input data into a text file.



File - C:\Users\camer_PythonClass\Assignment06\main.py

```
2 # Title: Assignment 06
3 # Description: Working with functions in a class,
4 #
              When the program starts, load each "row" of data
5 # in "ToDoList.txt" into a python Dictionary.
             Add the each dictionary "row" to a python list "table"
7 # ChangeLog (Who, When, What):
8 # RRoot, 1.1.2030, Created started script
9 # RRoot, 1.1.2030, Added code to complete assignment 5
10 # CMathews, 8.15.2021, Modified code to complete assignment 6
11 # CMathews, 8.17.2021, Modified output formatting
12 # ------ #
13
14 # Data ----- #
15 # Declare variables and constants
16 strFileName = "ToDoList.txt" # The name of the data file
17 objFile = None # An object that represents a file
18 dicRow = {} # A row of data separated into elements of a dictionary {Task,Priority}
19 lstTable = [] # A list that acts as a 'table' of rows
20 strChoice = "" # Captures the user option selection
21 strTask = "" # Captures the user task data
22 strPriority = "" # Captures the user priority data
23 strStatus = "" # Captures the status of an processing functions
24
25
26 # Processing -------- #
27 class Processor:
      """ Performs Processing tasks """
29
30
      @staticmethod
31
      def read_data_from_file(file_name, list_of_rows):
32
         """ Reads data from a file into a list of dictionary rows
33
         :param file_name: (string) with name of file:
34
35
         :param list_of_rows: (list) you want filled with file data:
36
         :return: (list) of dictionary rows
37
         list_of_rows.clear() # clear current data
38
```

```
file = open(file_name, "r")
39
40
           for line in file:
41
               task, priority = line.split(",")
42
               row = {"Task": task.strip(), "Priority": priority.strip()}
43
               list_of_rows.append(row)
44
           file.close()
45
           return list_of_rows, '\nSuccess.'
46
47
       @staticmethod
       def add_data_to_list(task, priority, list_of_rows):
48
49
           """Adds a task and its priority to the list of dictionary rows
50
           :param task: (string) with name of task:
51
52
           :param priority: (string) with priority of task:
53
           :param list_of_rows: (list) data you want added to list
54
           :return: (list) of dictionary rows
           11 11 11
55
56
           dicRow = {"Task": task, "Priority": priority}
57
           list_of_rows.append(dicRow)
58
           return list_of_rows, '\nSuccess - the task has been added.'
59
60
       @staticmethod
61
       def remove_data_from_list(task, list_of_rows):
62
           """Removes a task and its priority from the list
63
           :param task: (string) of task to remove
64
65
           :param list_of_rows: (list) of dictionary rows
66
           :return: (list) of dictionary rows, (string) message
67
68
           for item in list_of_rows:
               if task == item["Task"]:
69
70
                   list_of_rows.remove(item)
71
                   return list_of_rows, "\nSuccess - the task has been removed."
72
               else:
73
                   return list_of_rows, '\nError - task not found.'
74
75
       @staticmethod
       def write_data_to_file(file_name, list_of_rows):
76
```

```
"""Writes dictionary data to text file
 78
 79
           :param: file_name: (object) of text file
 80
           :param: list_of_rows: (list) of dictionary rows
81
           :return: (list) of dictionary rows, (string) message
 82
           objFileText = open(file_name, "w")
 83
 84
           for item in list_of_rows:
               objFileText.write(item["Task"] + ", " + item["Priority"] + "\n")
 85
 86
           objFileText.close()
 87
           return list_of_rows, '\nSuccess - the data has been saved.'
 88
 89
90 # Presentation (Input/Output) ----- #
 91 class IO:
       """ Performs Input and Output tasks """
 92
 93
 94
       @staticmethod
 95
       def print_menu_Tasks():
           """ Display a menu of choices to the user
 96
 97
 98
           :return: nothing
 99
           print('''
100
101
           Menu of Options
           1) Add a new Task
102
103
           2) Remove an existing Task
           3) Save Data to File
104
105
           4) Reload Data from File
106
           5) Exit Program
107
           ''')
108
           print() # Add an extra line for looks
109
110
       @staticmethod
111
       def input_menu_choice():
           """ Gets the menu choice from a user
112
113
114
           :return: string
```

```
115
           choice = str(input("Which option would you like to perform? [1 to 5] - ")).strip()
116
           print() # Add an extra line for looks
117
118
           return choice
119
120
       @staticmethod
121
       def print_current_Tasks_in_list(list_of_rows):
            """ Shows the current Tasks in the list of dictionaries rows
122
123
124
           :param list_of_rows: (list) of rows you want to display
125
           :return: nothing
126
           11 11 11
127
           print("\n****** The current Tasks ToDo are: ******\n")
128
           for row in list_of_rows:
129
               print(row["Task"] + " (" + row["Priority"] + ")")
130
           131
           print() # Add an extra line for looks
132
133
       @staticmethod
134
       def input_yes_no_choice(message):
135
           """ Gets a yes or no choice from the user
136
137
           :return: string
138
139
           return str(input(message)).strip().lower()
140
141
       @staticmethod
       def input_press_to_continue(optional_message=''):
142
143
           """ Pause program and show a message before continuing
144
145
           :param optional_message: An optional message you want to display
146
           :return: nothing
           11 11 11
147
148
           print(optional_message)
149
           input('\nPress the [Enter] key to continue.')
150
151
       @staticmethod
       def input_new_task_and_priority():
152
```

```
task = input("Task name: ")
153
           priority = input("Task priority: ")
154
155
           return task, priority
156
157
       @staticmethod
158
       def input_task_to_remove():
           task = input("What task would you like to remove? ")
159
160
           return task
161
162
164
165 # Step 1 - When the program starts, Load data from ToDoFile.txt.
166 Processor.read_data_from_file(strFileName, lstTable) # read file data
167
168 # Step 2 - Display a menu of choices to the user
169 while (True):
170
       # Step 3 Show current data
171
       IO.print_current_Tasks_in_list(lstTable) # Show current data in the list/table
172
       IO.print menu Tasks() # Shows menu
173
       strChoice = I0.input_menu_choice() # Get menu option
174
       # Step 4 - Process user's menu choice
175
176
       if strChoice.strip() == '1': # Add a new Task
177
           strTask, strPriority = I0.input_new_task_and_priority()
           lstTable, strStatus = Processor.add_data_to_list(strTask, strPriority, lstTable)
178
179
           IO.input_press_to_continue(strStatus)
180
           continue # to show the menu
181
       elif strChoice == '2': # Remove an existing Task
182
183
           strTask = I0.input_task_to_remove()
184
           lstTable, strStatus = Processor.remove_data_from_list(strTask, lstTable)
185
           IO.input_press_to_continue(strStatus)
186
           continue # to show the menu
187
188
       elif strChoice == '3': # Save Data to File
189
           strChoice = I0.input_yes_no_choice("Save this data to file? (y/n) - ")
           if strChoice.lower() == "v":
190
```

File - C:\Users\camer_PythonClass\Assignment06\main.py

```
191
                lstTable, strStatus = Processor.write_data_to_file(strFileName,lstTable)
192
                IO.input_press_to_continue(strStatus)
193
            else:
194
                IO.input_press_to_continue("Save Cancelled!")
195
            continue # to show the menu
196
197
        elif strChoice == '4': # Reload Data from File
198
            print("Warning: Unsaved Data Will Be Lost!")
199
            strChoice = I0.input_yes_no_choice("Are you sure you want to reload data from file? (y/n) - ")
            if strChoice.lower() == 'y':
200
201
               lstTable, strStatus = Processor.read_data_from_file(strFileName,lstTable)
202
               IO.input_press_to_continue(strStatus)
203
            else:
204
                IO.input_press_to_continue("File Reload Cancelled!")
205
            continue # to show the menu
206
207
        elif strChoice == '5': # Exit Program
208
            print("Goodbye!")
209
            break # and Exit
210
```

Appendix B

```
1 C:\Users\camer\AppData\Local\Programs\Python\Python39\python.exe C:/Users/camer/_PythonClass/Assignment06/main.py
2
3 ***** The current Tasks ToDo are: *****
5 Walk Dog (Medium)
6 Laundry (High)
 8 **********
10
11
          Menu of Options
12
          1) Add a new Task
13
          2) Remove an existing Task
          3) Save Data to File
14
15
          4) Reload Data from File
16
          5) Exit Program
17
19 Which option would you like to perform? [1 to 5] - 1
21 Task name: Groceries
22 Task priority: Low
23
24 Success - the task has been added.
26 Press the [Enter] key to continue.
28 ***** The current Tasks ToDo are: *****
29
30 Walk Dog (Medium)
31 Laundry (High)
32 Groceries (Low)
34 ************
35
36
37
          Menu of Options
38
          1) Add a new Task
```

```
39
          2) Remove an existing Task
40
          3) Save Data to File
41
          4) Reload Data from File
42
          5) Exit Program
43
45 Which option would you like to perform? [1 to 5] - 2
47 What task would you like to remove? Walk Dog
49 Success - the task has been removed.
51 Press the [Enter] key to continue.
53 ***** The current Tasks ToDo are: *****
55 Laundry (High)
56 Groceries (Low)
58 ************
59
60
61
          Menu of Options
62
          1) Add a new Task
63
          2) Remove an existing Task
64
          3) Save Data to File
65
          4) Reload Data from File
          5) Exit Program
67
69 Which option would you like to perform? [1 to 5] - 3
71 Save this data to file? (y/n) - y
72
73 Success - the data has been saved.
75 Press the [Enter] key to continue.
```

```
77 ***** The current Tasks ToDo are: *****
78
79 Laundry (High)
80 Groceries (Low)
81
 82 ***********
83
84
85
          Menu of Options
          1) Add a new Task
86
87
          2) Remove an existing Task
88
          3) Save Data to File
89
          4) Reload Data from File
90
          5) Exit Program
91
92
93 Which option would you like to perform? [1 to 5] - 4
94
95 Warning: Unsaved Data Will Be Lost!
96 Are you sure you want to reload data from file? (y/n) - y
 97
98 Success.
100 Press the [Enter] key to continue.
102 ***** The current Tasks ToDo are: *****
103
104 Laundry (High)
105 Groceries (Low)
106
107 ************
108
109
110
          Menu of Options
111
          1) Add a new Task
112
          2) Remove an existing Task
113
          3) Save Data to File
          4) Reload Data from File
114
```

```
115
           5) Exit Program
116
117
118 Which option would you like to perform? [1 to 5] - 5
119
120 Goodbye!
121
122 Process finished with exit code 0
123
```