Moiz Vahora

08/02/2021

IT FDN 110 A

Assignment 4

**Working with Dictionaries and Files**

**Introduction**

The script made for this assignment reads a file called “TodDoList.txt” with a list of tasks and priorities and processes, displays, edits the dataset in the file using a menu of options provided to the user by the program. The file is then updated with the user modifications if they choose to save their edits to file. This assignment requires the user to modify a template provided by the instructor.

**Steps Taken**

These are the step-by-step tasks taken by the author of the code to create the program used to load, edit, and save the ToDoList.txt file. This is broken down into 3 sections consisting of a data section which is used to initialize the variables used in the program, a processing section to load the data from the ToDoList.txt file, and a input/output section used to display, modify, and save the data loaded from the ToDoList.txt file. Details are elaborated bellow:

1. DATA
   1. This section declares the variables and constants used in the program. Its ideal to declare the variables and their type in advance to avoid issues with them as data is being processed in the program.
2. PROCESSING
   1. This section loads the data from the ToDoList.txt file and adds it to a list containing a “table” of rows. This is done by reading the file using the open function and using a for loop to add each row to a list variable called 1stRow, which is split into two strings using the comma as a divider. Each input in the list is inputted into the dictionary variable dicRow, which has the identifiers "Task” and “Priority”. This is dicRow variable for each loop is then appended into the list 1stTable variable for use in the next section of the program.
3. INPUT/OUTPUT

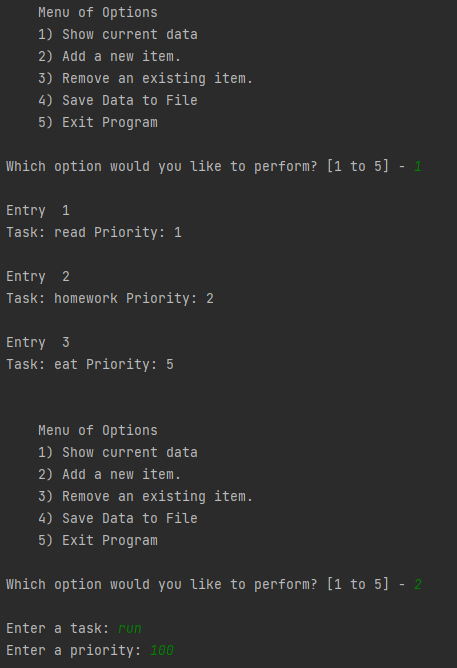
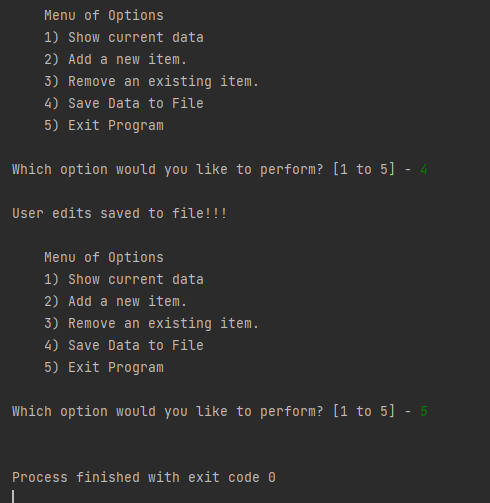
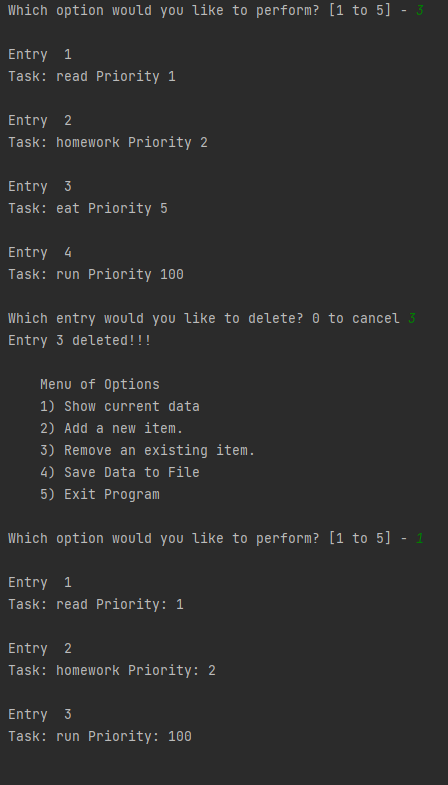
The following options repeated inside of a while loop until the user exits the program by selecting option 5.

* 1. Displaying Menu of options to user
     1. This section displays the options for the user for the list used in the data set. The input by the user is stored in a variable strChoice and is used to select the options bellow.
  2. Show current data set
     1. This is option 1 and a for loop is used to print the dictionary entries in each “row” of the list variable. A counter is also used to display the entry number for each row within the list variable.
  3. Adding new item to list/table
     1. This is option 2 and it asks to input a new “Task” and a its “Priority”, this information is then added to the dicRow variable and is appended to the 1stTable list variable.
  4. Deleting item from list/table
     1. This is option 3 and it displays the current data set using the same script used in option 1. the user is then prompted to selected which entry they would like to delete by inputing the entry number (0 if they would like to cancel) and the remove function is used to delete the entry from the list. A message then displays the entry number delted by the user.
  5. Saving tasks to the ToDoList.txt file
     1. This is option 4 and the program opens the ToDoList.txt file and overites all the information in the file using the “w” command in the open function. This is used in case the user wanted to delete the entries from the original file as opposed to appending and retaining the original data. A for loop is used to write the each row entry from list into the ToDoList.txt file, where each row entry is a dictionary entry, which each string is separated using a comma along with a new row at the end. The file is then closed a message is printed to inform the user their edits were saved.
  6. Exit program
     1. This is option 5 and the loop is broken using a break, causing the program to exit.

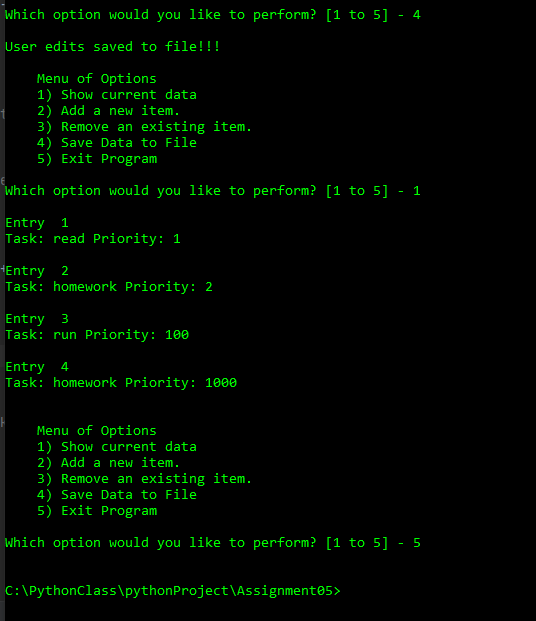
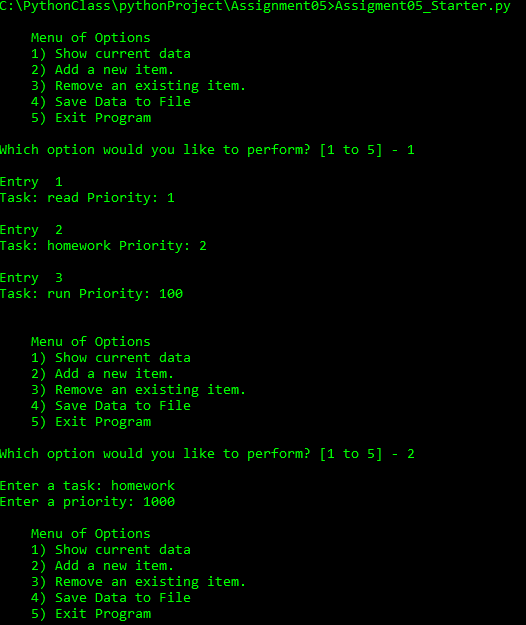
**Python Code**

# -----------------------------  
# ------------------------------------------------------------------------ #  
# Title: Assignment 05  
# Description: Working with Dictionaries and Files  
# When the program starts, load each "row" of data  
# in "ToDoToDoList.txt" into a python Dictionary.  
# Add the each dictionary "row" to a python list "table"  
# ChangeLog (Moiz Vahora, 8/10/21,Modified code to look cleaner):  
# RRoot,08.06.2021,Created started script  
# <Moiz Vahora>,<08.10.2021>,Added code to complete assignment 5  
# ------------------------------------------------------------------------ #  
  
# -- Data -- #  
# declare variables and constants  
objFile = "ToDoList.txt" # An object that represents a file  
strData = "" # A row of text data from the file  
dicRow = {} # A row of data separated into elements of a dictionary {Task,Priority}  
lstTable = [] # A list that acts as a 'table' of rows  
# strMenu didn't need to be used so I commented it out. Though I can define it as the menu option listed in the code  
# strMenu = "" # A menu of user options  
strChoice = "" # A Capture the user option selection  
  
# -- Processing -- #  
# Step 1 - When the program starts, load the any data you have  
# in a text file called ToDoList.txt into a python list of dictionaries rows (like Lab 5-2)  
# *TODO: Add Code Here*  
*# Process the data*  
*objFile\_IO =* open(objFile, "r") # open file for reading  
  
# Format data from file into a list that is processed by user  
for row in objFile\_IO:  
 lstRow = row.split(",")  
 dicRow = {"Task": lstRow[0], "Priority": lstRow[1]}  
 lstTable.append(dicRow)  
objFile\_IO.close() # closing file  
  
# -- Input/Output -- #  
# Step 2 - Display a menu of choices to the user  
while (True):  
 print("""  
 Menu of Options  
 1) Show current data  
 2) Add a new item.  
 3) Remove an existing item.  
 4) Save Data to File  
 5) Exit Program  
 """)  
 strChoice = str(input("Which option would you like to perform? [1 to 5] - "))  
 print() # adding a new line for looks  
 # Step 3 - Show the current items in the table  
 if (strChoice.strip() == '1'):  
 # *TODO: Add Code Here*  
for i\_entry, objRow in enumerate(lstTable): #using i\_entry to display index of list for user reference (added 1 since index starts at 0)  
 print("Entry ", i\_entry + 1)  
 print('Task: ' + objRow["Task"] + ' ' + 'Priority: ' + objRow["Priority"])  
  
 continue  
  
 # Step 4 - Add a new item to the list/Table  
 elif (strChoice.strip() == '2'):  
 # *TODO: Add Code Here*  
# Ask for user input  
 I\_task = input("Enter a task: ")  
 I\_priority = input("Enter a priority: ")  
 # appending list variable with user input using a dictionary  
 dicRow = {"Task": I\_task, "Priority": I\_priority + '\n'}  
 lstTable.append(dicRow)  
  
 # Step 5 - Deleting item from the list/Table  
 elif (strChoice.strip() == '3'):  
 # *TODO: Add Code Here*  
for i\_entry, objRow in enumerate(lstTable):  
 # diplaying current data for user reference (copied from option 1)  
 print("Entry ", i\_entry + 1)  
 print('Task: ' + objRow["Task"] + ' ' + 'Priority ' + objRow["Priority"])  
 #Asking user which entry to delete (0 to cancel)  
 delete = int(input("Which entry would you like to delete? 0 to cancel "))  
  
 if delete == 0:  
 # cancel deletion and return to menu  
 print("No user input deleted")  
 else:  
 # Removing selected entry (subtract 1 since count starts at 0) and displaying entry number deleted  
 lstTable.remove(lstTable[delete - 1])  
 print("Entry " + str(delete) + " deleted!!!")  
 continue  
  
 continue  
 # Step 6 - Save tasks to the ToDoToDoList.txt file  
 elif (strChoice.strip() == '4'):  
 # *TODO: Add Code Here*  
# Opening file and overwriting my previous input, in case we deleted entries  
 objFile\_IO = open(objFile, "w")  
 for objRow in lstTable:  
 # Writing entries in list to the file from each dictionary entry  
 objFile\_IO.write(objRow["Task"] + ',' + objRow["Priority"])  
 objFile\_IO.close() # close file  
 print("User edits saved to file!!!")  
  
 # Step 7 - Exit program  
 elif (strChoice.strip() == '5'):  
 # *TODO: Add Code Here*  
# nothing to add program loop is broken and the program exits  
 break # and Exit the program

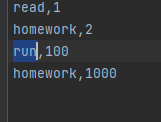
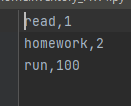
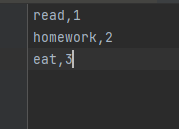
**PyCharm Display**

**Command Prompt Display**



**ToDoList.txt File (results from PyCharm then CMD)**



**Conclusion**

The script written for this assignment is able to take data from the ToDoList.txt file and display/manipulate the data using dictionaries and lists. The overall program was relatively simple, though getting used to handling dictionaries was among the challenges as it is very different compared to C or MATLAB.