Gibran D. Washington

Professor Randal Root

ITN FDN 100: Assignment06

12 November 2016

User Program to do list, Voodoo ToDo

I utilized the solution that was given for Assignment05, from Professor Root as the basis for completing the coding project. The beginning of the program which opened the read file remaining unchanged with some minor variable changes in the dictionary table named ***taskRow***. Figure 1 has the short section of code that opens file to read, declares variables, and closes file before methods are loaded into memory in Figure 2.

objTDLog = "C:\\\_PythonClass\\ToDo1.txt"  
#-- Data --#  
# declare variables and constants  
# strData = A row of text data from the file  
strData = ""  
# dictList = A row of data separated into elements of a dictionary {Task,Priority}  
dictList = {}  
# taskRow = A dictionary that acts as a 'table' of rows  
taskRow = []  
  
  
# objTDL = An object that represents a file  
objTDL = open(objTDLog, "r")  
**for** line **in** objTDL:  
 strData = line.split(",") # readline() reads a line of the data into 2 elements  
 dictList = {"Task":strData[0].strip(), "Priority":strData[1].strip()}  
 taskRow.append(dictList)  
objTDL.close()

Figure Data, declared variables, and the read text

The function ***menuChoice()*** was defined and included all print statements strings to display the menu choice and not included in the class method. ***TaskOrganizer*()** has loaded with the code from Assignment05 solutions into three functions: ***AddTask()****,* ***RemoveTask*()**, and ***ShowAllTask()****.*

# Display a menu of choices to the user  
**def menuChoice**():  
 *'''function displays the menu options for users throughout the script'''* print(  
 """  
 Voodoo ToDo Menu Options:  
  
 Press 0 - Exit without saving.  
 Press 1 - Add a new task.  
 Press 2 - Remove an existing task.  
 Press 3 - Save tasks to file and exit!  
 Press 4 - Show current task  
 """)  
  
# a class for the methods that correspond to menu selection 1, 2, 3 and 4  
**class TaskOrganizer**():  
 *'''This class has 3 methods for editing the ToDo table list based on the menu selections'''* # define method 1, add a task to table  
 @staticmethod  
 **def AddTask**():  
 *''' this method allows user to add to the task list, appending to the dictionary'''* strTask = str(input("What is the task? - ")).strip()  
 strPriority = str(input("What is the priority? [high|medium|low] - ")).strip()  
 dictList = {"Task": strTask, "Priority": strPriority}  
 taskRow.append(dictList)  
  
 # define method 2, remove a task from table  
 @staticmethod  
 **def RemoveTask**():  
 *'''removes a task selected through looking at every task and matching with user input'''* strKeyToRemove = input("Which Task Would You Like to Remove? : ")  
 blnItemRemoved = **False** # Creating a boolean Flag  
 intRowNumber = 0  
 **while** (intRowNumber < len(taskRow)):  
 # the values function creates a list!  
 **if** (strKeyToRemove == str(list(dict(taskRow[intRowNumber]).values())[1])):  
 **del** taskRow[intRowNumber]  
 blnItemRemoved = **True** # end if  
 intRowNumber += 1  
 **if** blnItemRemoved == **True**:  
 print("The task was removed.")  
 **else**:  
 print("I'm sorry, but I could not find that task.")  
 print("\*\*\*\*\*\*\* The current task in ToDo are: \*\*\*\*\*")  
  
 # function shows the list of current task  
 @staticmethod  
 **def ShowAllTask**():  
 *'''function shows all of the task that are currently in the dictionary table'''* **for** row **in** taskRow:  
 print(row["Task"] + "(" + row["Priority"] + ")")

Figure Class, and method/functions for performing user request from menu, user menu display is not included in class, TaskOrganizer but is defined above in block

I thought I might need to declare a global variable but remove the code upon realizing it wasn’t necessary since the program is not doing any calculations that get used throughout the program and was easier to allow the print statements to be encapsulated with the functions.

# -- Input/Output --#  
**while**(**True**):  
 # Display a menu of choices to the user  
 menuChoice()  
 # strSelect = Capture the user option selection  
 strSelect = int(input("Please Make a Selection to begin [0 to 4]: "))  
  
 #-- Processing --#  
 # Show the current items in the table  
 **if** strSelect == 4:  
 print("\*\*\*\*\*\*\* The current task in ToDo are: \*\*\*\*\*")  
 TaskOrganizer.ShowAllTask()  
 print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")  
  
 # Add a new item to the list/Table  
 **elif** strSelect == 1:  
 TaskOrganizer.AddTask()  
 print("Current task in table after adding data are:")  
 TaskOrganizer.ShowAllTask()  
 print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")  
  
 # remove item from the list/table  
 **elif** strSelect == 2:  
 TaskOrganizer.RemoveTask()  
 print("\*\*\*\*\*\*\* The current task in ToDo are: \*\*\*\*\*")  
 TaskOrganizer.ShowAllTask()  
 print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")  
 # Display a menu of choices to the user  
 menuChoice()  
  
  
 # Show the current items in the table  
 **elif** strSelect == 3:  
 print("\*\*\*\*\*\*\* The current task in ToDo are: \*\*\*\*\*")  
 TaskOrganizer.ShowAllTask()  
 print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")  
 # Ask if they want save that data  
 **if** ("y" == str(input("Save this data to file? (y/n) - ")).strip().lower()):  
 objTDL = open(objTDLog, "a")  
 **for** dicRow **in** taskRow:  
 objTDL.write(dicRow["Task"] + "," + dicRow["Priority"] + "\n")  
 objTDL.close()  
 input("Data saved to file! Press the [Enter] key to return to menu.")  
 **else**:  
 input("New data was NOT Saved, but previous data still exists! Press the [Enter] key to return to menu.")  
 # Display a menu of choices to the user  
 menuChoice()  
  
  
 **elif** strSelect == 0:  
 # Exit the program  
 **break**

Figure Main body of program with loop, with if and 4 elif statements

C:\Python34\python.exe C:/\_PythonClass/Module6Project/Assignment06.py

##############################################################################

Voodoo ToDo

This Program is designed to help end the bad habits of not writing down tasks or completing

tasks based on the level of importance you assign. It stores your task list in a file for portability.

##############################################################################

Voodoo ToDo Menu Options:

Press 0 - Exit without saving.

Press 1 - Add a new task.

Press 2 - Remove an existing task.

Press 3 - Save tasks to file and exit!

Press 4 - Show current task

Please Make a Selection to begin [0 to 4]: 1

What is the task? - Buy Groceries

What is the priority? [high|medium|low] - medium

Current Task in table After Adding Data is:

Clean House(low)

Pay Bills(high)

Buy Groceries(medium)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Figure Sample Output after adding a task to the existing dictionary list

Above in Figure 4, a output run of the program with a user selection of 1 which shows the results of two methods being called to add a task and display the current task list following an addition of a task. The only other modifications employed were that of a ***menuChoice()*** function added at the end of the while loop instead of a continue that would send control back to the beginning of the loop to display the menu. I removed this function call from the final version of program. I also changed the parameter for saving the file to an “a” instead of the “w”.

Works Cited

Dawson, Michael. Python Programming for the Absolute Beginner, 3rd ed. Boston: Cengage Learning PTR, 2011.

Learn Python

<http://www.learnpython.org/en/Functions>. World Wide Web. Retrieved 4 November 2016.

Payne, Trevor. Let’s Learn Python – Basics #6 of 8 – Functions, *YouTube, uploaded by Trevor*

*Payne,* 4 November 2016.<https://youtu.be/qO4ZN5uZSVg>. Retrieved 4 November 2016.

Root, Randal. Python Videos: Python Module 06-01 to 06-13, *YouTube, uploaded by Randal*

*Root,* 13 October 2016.

Tutorials Point Simply Easy Learning

<http://www.tutorialspoint.com/python/python_functions.htm>. World Wide Web. Retrieved

4 November 2016