Inderjot Brar

February 16, 2021

IT FDN 110 B

Assignment 05

ASSIGNMENT 5

**Introduction**

This assignment documents the steps I took to create and test a Python Script that reads data from a text file and allows user to modify the data. The script uses dictionary and list objects to load, store and modify the data from the text files.

## **Creating the Script**

First, I created a ‘ToDoList’ text file, containing 2 types of data – ‘Task’ and ‘Priority’. I modified the ‘Assignment05\_Starter’ script to add the python code needed to perform the processing. I started off by updating the change log and then declaring the variables that are to be used in the script in one block at the top to better organize the ode (Figure 1)

Text

Description automatically generated

***Figure 1: Script Header and Script Variable***

Next, I used ‘open’ function to read the data from the text file and used ‘for’ statement to load each row of data from the file. Data from each row was first added to a list, dictionary rows were created by assigning keys to values in the list and added to a list table by using ‘append’ method. (Figure 2)

Graphical user interface, text, application, email

Description automatically generated

***Figure 2: Code added to read data from text file***

The code uses ‘while’ loop to provide users with a menu of choices to list, modify or save data to the file and continues to run until the user chooses to exit, causing a break in the loop. ‘If-elif’ statements are used to execute the code lines specific to the user choice. For option 1, the code uses ‘for’ loop and ‘print’ function to display each row in the list table created under the processing section of the script.

**Graphical user interface, text, application

Description automatically generated**

***Figure 3: Code added to display data from file to the user***

For option 2, ‘input’ function is used to capture new data from the user, which is then stored in a new dictionary row and added to the list table. For option 3, ‘print’ function and ‘for’ loop is used to provide user with a list of current tasks and ‘input’ function is used to capture the task user wants to remove. Within a ‘for’ loop, if function is used to match the user input with the values stored under Task key in each row and remove the row from the list. An empty variable ‘countRemove’ is initialized before running the loop and its value is increased by 1 each time the loop is run. If at the end, the variable still has a value of 0, the user is informed that the task they entered is not in the table.

**Graphical user interface, text, application, email

Description automatically generated**

***Figure 4: Code added to allow user to add a new item or remove an existing item***

For option 4, the text file is opened in write mode and ‘for’ loop is used to write each row of data from the list table. For option 5, break statement is used to allow user to exit the program.

**Graphical user interface, text

Description automatically generated**

***Figure 5: Code added to save updated data to the file and exit program***

## **Testing the Script in PyCharm**

After saving the script, I ran the script in PyCharm to test my code. I successfully entered a task and its priority and saved the data in the text file before exiting. (Figure 6)

Text

Description automatically generated

***Figure 6: Script module successfully tested in PyCharm.***

## **Running the Script in OS Command**

Next, I ran the script from the Terminal on my Mac to ensure that it works properly. I was able to successfully test removing a task from the file before saving it. (Figure 7)

Text

Description automatically generated

***Figure 7: Script executed from Terminal on Mac OS.***

## **Verifying text file**

Lastly, I located the text file and verified the data was updated properly. During testing, I had input data with varying cases. As all the tasks were formatted similarly in the text file, my formatting had worked as desired. (Figure 5)

Text

Description automatically generated with medium confidence

***Figure 5: Verification of HomeInventory.txt file.***

## **Summary**

Using the Module 05 instructions, I was able to successfully create and run the script to read and modify data from a text file. The program demonstrates my knowledge about creating dictionaries and lists, using while and for loops and if-elif statements, and writing data into a text file.