Name: Natalie Pritchett

Date: 8/9/2023

Course: Foundations of Programming: Python

Assignment05 – Working with Dictionaries and Files

Introduction and TODO #1

To get started I opened the Assignment05_Starter.py file using the Visual Studio Code editor. Then, I edited the script header to update the change log. I modified the read portion of the code from Lab 5-2 so that each row of data that had already been saved to ToDoList.txt file would be loaded into a dictionary. The dictionary keys create the columns, so that each row can be added to a List. I appended each row to create a table of the previously saved data. I also included a try and except statement in case no file had been created yet.

```
23  # Step 1 - When the program starts, load the any data you have
24  # in a text file called ToDoList.txt into a python list of dictionaries rows (like Lab 5-2)
25  # TODO: Add Code Here
26  # File to List
27  try:
28      objFile = open(objFile, "r") # read in the data from the text file
29      for row in objFile: # to loop through the rows of data
30          lstRow = row.split(",")
31          dicRow = {"Task": lstRow[0], "Priority": lstRow[1].strip()}
32          lstTable.append(dicRow) # add the rows to the table in a list
33          print (lstTable)
34      objFile.close()
35  except:
36      print("No file yet")
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS C:\Users\natpri\OneDrive - UW\Documents\Training\Python\Module05 - Lists and Dictionaries\Assign
ython.exe "c:/Users/natpri/OneDrive - UW/Documents/Training/Python/Module05 - Lists and Dictionarie
y"
[{'Task': 'mop', 'Priority': 'high'}]
[{'Task': 'mop', 'Priority': 'high'}, {'Task': 'sweep', 'Priority': 'low'}]
```

TODO #2

The next to do item was to show the current items in the table when option 1 is selected. Since the dictionary of rows had already been added to a list, I created a for loop to print each row.

```
50      # Step 3 - Show the current items in the table
51      if (strChoice.strip() == '1'):
52          # TODO: Add Code Here
53          for row in lstTable: # go through the list table
54              print(row) # show the current data
55          continue
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

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

{'Task': 'mop', 'Priority': 'high'}
{'Task': 'sweep', 'Priority': 'low'}
```

TODO #3

If the user selects option 2 they will be prompted to:

Enter a task and then assign it a priority level.

Each row is then added to the dictionary of existing data and then appended to the list to print.

```
56      # Step 4 - Add a new item to the list/Table
57      elif (strChoice.strip() == '2'):
58          Task = input("Enter an Task: ") # add a new task
59          Priority = input("Enter a Priority: ") # add the priority level
60          dicRow = {"Task":Task, "Priority":Priority} # row of data
61          lstTable.append (dicRow) # each dicRow is part of a table data
62          print (lstTable) # see the newest addition to the table list
63          continue
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                    Python

Enter an Task: mow the lawn
Enter a Priority: low
[{'Task': 'mop', 'Priority': 'high'}, {'Task': 'sweep', 'Priority': 'low'}, {'Task': 'mow the lawn', 'Priority': 'low'}]
```

TODO #4

Choosing option 3 removes the last item that was entered from the list and prints the new list without that item entry.

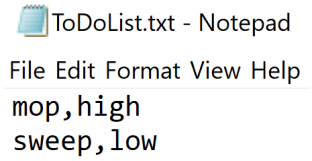
```
64      # Step 5 - Remove a new item from the list/Table
65      elif (strChoice.strip() == '3'):
66          # slicing the list
67          lstTable= lstTable[:-1]
68          # print the updated list
69          print("New list: ",lstTable)
70          continue
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Which option would you like to perform? [1 to 5] - 3

New list:  [{'Task': 'mop', 'Priority': 'high'}, {'Task': 'sweep', 'Priority': 'low'}]
```

TODO #5

If the user selects option 3, the user entered information is then written to the text file 'ToDoList.txt'. The print statement lets the user know where the data entered for the program was saved. The data entered can be verified by reviewing the saved text file.

Verifying that the file saved correctly:



```
71      # Step 6 - Save tasks to the ToDoToDoList.txt file
72      elif (strChoice.strip() == '4'):
73          objFile = open("ToDoList.txt", "w")
74          for row in lstTable:
75              objFile.write(str(row["Task"]) + ',' + str(row["Priority"] +'\n'))
76          objFile.close()
77          print('Data was saved')
78          continue
```

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

Which option would you like to perform? [1 to 5] - 4

Data was saved
```

Conclusion

The last option ends the program and the print out lets the user know that the program is exiting.

```
79      # Step 7 - Exit program
80      elif (strChoice.strip() == '5'):
81          print('Exiting Program')
82          break  # and Exit the program
83      print('please choose from the options from 1 to 5')
84
```

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

Exiting Program
```