Nathan Shelby
10MAY2023
Foundations of Programming: Python
Assignment 05
https://github.com/Nebulis01/IntroToProg-Python

Assignment 05: To Do List

Introduction:

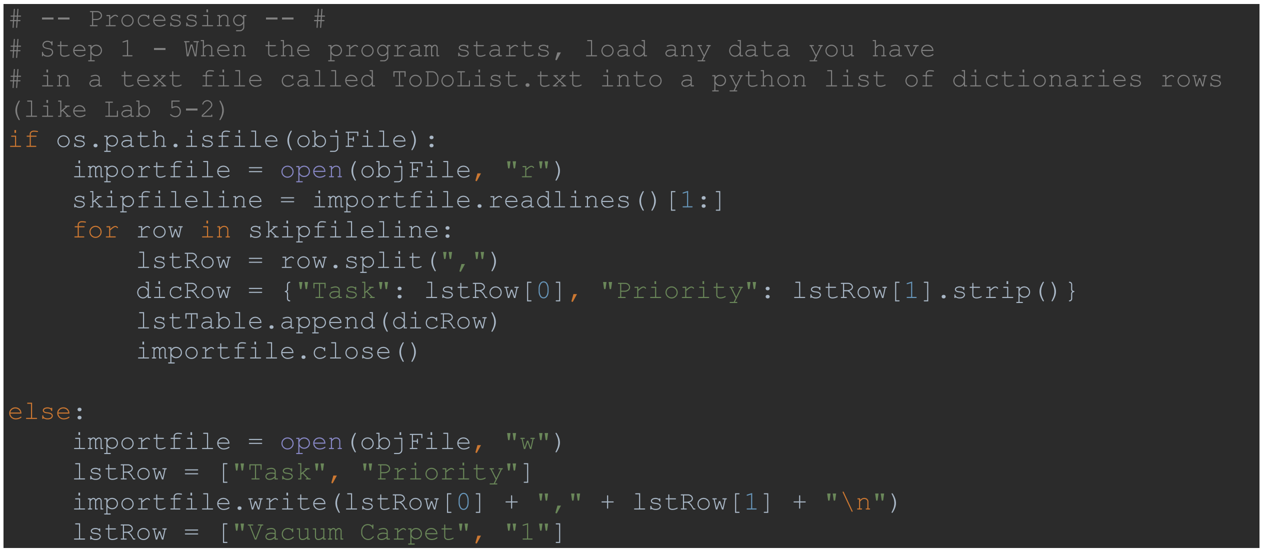
This is the 5th assignment, part of Module 5 performed as part of the Foundations of Programming: Python course for the Spring 2023 session at University of Washington. The task list includes reading chapter 5 of the coursebook and several web videos covering reading files, writing files, dictionary, and list manipulation within Python.

The primary coursework this week is to modify a starting program to accomplish several tasks including reading data from a file, echoing it back to the user, manipulating data in memory and writing it to disk – a core requirement of the ask was *not* to utilize any *functions* within the code.

Creating your script:

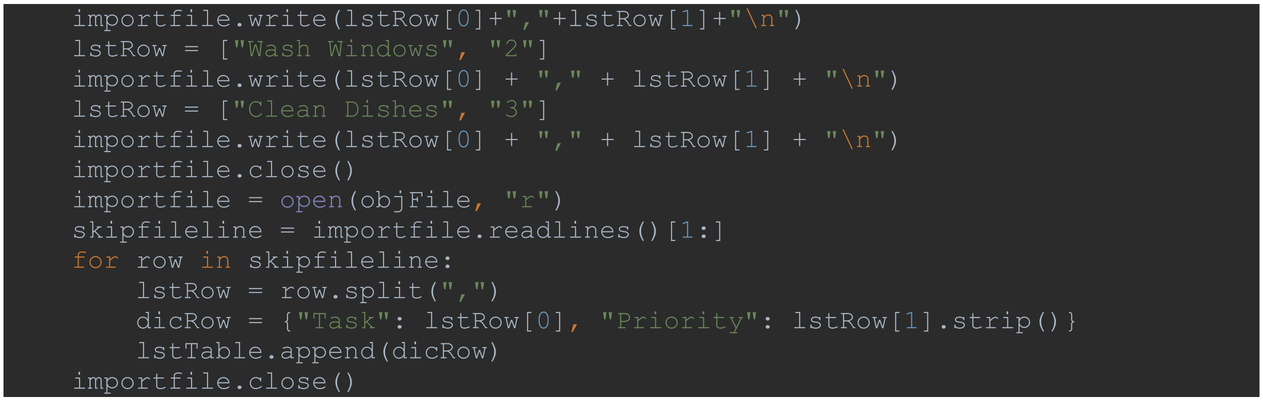
In this script we were asked to use the provided starter and work inside to accommodate the asks within – read data from a file (if present)? Use the menu and manipulate the data in a number of ways – display, edit, write.

Running through main sets up sample data. We look at the disk to see if *ToDoList.*txt is present, importing data (skipping the header row) if it is. Otherwise, the *else* statement creates both the sample data in *lstTable* and writes data to disk for a subsequent run. [Figure 1]



```
# -- Processing -- #
# Step 1 - When the program starts, load any data you have
# in a text file called ToDoList.txt into a python list of dictionaries rows
(like Lab 5-2)
if os.path.isfile(objFile):
    importfile = open(objFile, "r")
    skipfileline = importfile.readlines()[1:]
    for row in skipfileline:
        lstRow = row.split(",")
        dicRow = {"Task": lstRow[0], "Priority": lstRow[1].strip()}
        lstTable.append(dicRow)
        importfile.close()

else:
    importfile = open(objFile, "w")
    lstRow = ["Task", "Priority"]
    importfile.write(lstRow[0] + "," + lstRow[1] + "\n")
    lstRow = ["Vacuum Carpet", "1"]
```



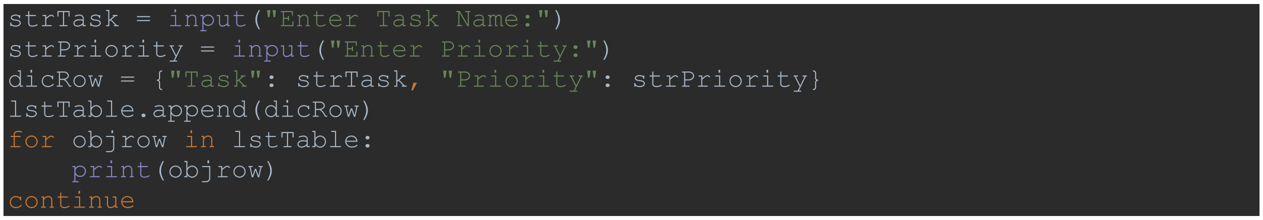
```
    importfile.write(lstRow[0]+","+lstRow[1]+"\n")
    lstRow = ["Wash Windows", "2"]
    importfile.write(lstRow[0] + "," + lstRow[1] + "\n")
    lstRow = ["Clean Dishes", "3"]
    importfile.write(lstRow[0] + "," + lstRow[1] + "\n")
    importfile.close()
    importfile = open(objFile, "r")
    skipfileline = importfile.readlines()[1:]
    for row in skipfileline:
        lstRow = row.split(",")
        dicRow = {"Task": lstRow[0], "Priority": lstRow[1].strip()}
        lstTable.append(dicRow)
    importfile.close()
```

Figure 1 – # -- Processing -- #

The menu makes use of a while() loop and once input is entered an if() statement evaluates the provided input for 1,2,3,4,5 and processes accordingly.

For a press of 1 a *for* loop is used to iterate through the *lstTable* list and display data to the user.

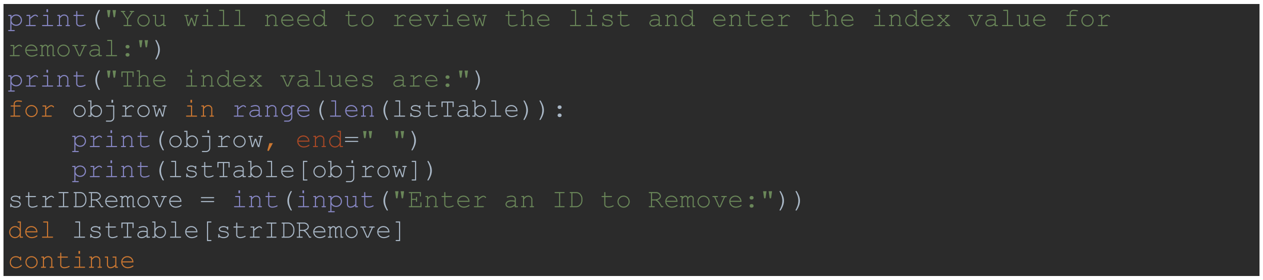
For a press of 2 (see Figure 2) ask the user to provide input in string form, add those strings to a dictionary and append the dictionary to the in memory *lstTable* list, echoing the content back after the addition.



```
strTask = input("Enter Task Name:")
strPriority = input("Enter Priority:")
dicRow = {"Task": strTask, "Priority": strPriority}
lstTable.append(dicRow)
for objrow in lstTable:
    print(objrow)
continue
```

Figure 2 – '2'

For a press of 3 (see figure 3) we instruct the user to review a printed list and provide a value to remove. We loop through the contents of *lstTable* and provide the location of items within the list to remove. We prompt the user to enter that integer input and pass that input to a *del* statement removing the item and kicking them back to the menu home.



```
print("You will need to review the list and enter the index value for
removal:")
print("The index values are:")
for objrow in range(len(lstTable)):
    print(objrow, end=" ")
    print(lstTable[objrow])
strIDRemove = int(input("Enter an ID to Remove:"))
del lstTable[strIDRemove]
continue
```

Figure 3 – '3'

For a press of 4 (see figure 4) we will save the data in *lstTable* to disk. We provide an echo of the contents of *lstTable* to the user and inform them that we will write to disk. As we've been using the *lstTable* list in memory we need to break this apart in to the *string* datatype for the *write()* or *writelines()* methods to handle. We do this by iterating through the contents of *lstTable* and creating a string variable with the dictionary items for *'Task'* and *'Priority'* that we have been visualizing throughout the program. We then pass that string variable to a file handler and when we have done this for all rows in *lstTable* we close the file so that python will commit the write to disk. Reporting back to the user that the write to disk was successful – this could be wrapped in a try/catch to improve resilience.

```
for objrow in lstTable:
    print(objrow)
print("Saving above list to file")
writefile = open(objFile, "w")
lstRow = ["Task", "Priority"]
writefile.write(lstRow[0] + "," + lstRow[1] + "\n")
for objrow in lstTable:
    strtowrite = objrow['Task'] + "," + objrow['Priority'] + "\n"
    writefile.writelines(strtowrite)
writefile.close()
print("Successfully saved to file.")
continue
```

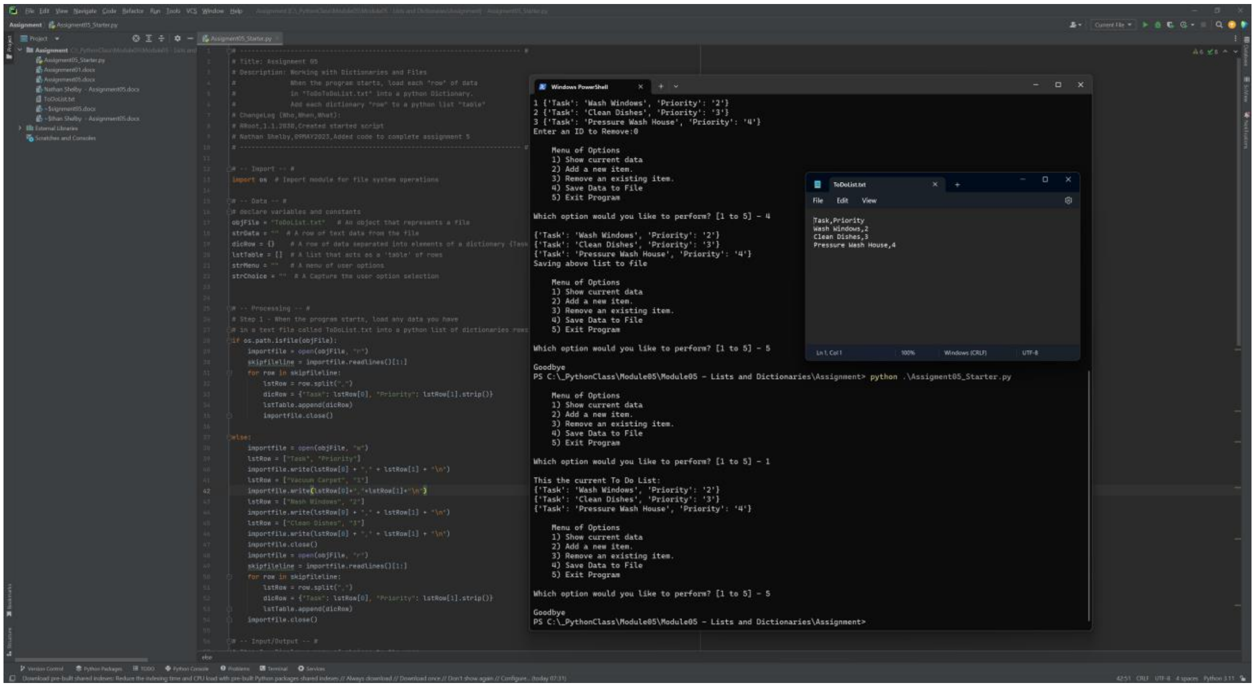
Figure 4 – '4'

For a 5 press we simply exit the program, printing a message because programs should also have manners 😊.

To provide proof of execution on the local client, ensuring that the script is functional a screenshot is provided. [See Figure 5]

Summary

This week we learned more about lists, working with dictionaries and basic file operations. I look forward to Module 6 where some of this is likely wrapped in functions for increased functionality.



```
# ------------------------------------------------------------------------ #
# 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 each dictionary "row" to a python list "table"
# ChangeLog (Who,When,What):
# RRoot,1.1.2030,Created started script
# Nathan Shelby,09MAY2023,Added code to complete assignment 5
# ------------------------------------------------------------------------ #

# -- Import -- #
import os  # Import module for file system operations

# -- 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
lstTable = []  # A list that acts as a 'table' of rows
strMenu = ""   # A menu of user options
strChoice = ""  # A Capture the user option selection


# -- Processing -- #
# Step 1 - When the program starts, load any data you have
# in a text file called ToDoList.txt into a python list of dictionaries rows
if os.path.isfile(objFile):
    importfile = open(objFile, "r")
    skipfileline = importfile.readlines()[1:]
    for row in skipfileline:
        lstRow = row.split(",")
        dicRow = {"Task": lstRow[0], "Priority": lstRow[1].strip()}
        lstTable.append(dicRow)
        importfile.close()

else:
    importfile = open(objFile, "w")
    lstRow = ["Task", "Priority"]
    importfile.write(lstRow[0] + "," + lstRow[1] + "\n")
    lstRow = ["Vacuum Carpet", "1"]
    importfile.write(lstRow[0]+","+lstRow[1]+"\n")
    lstRow = ["Wash Windows", "2"]
    importfile.write(lstRow[0] + "," + lstRow[1] + "\n")
    lstRow = ["Clean Dishes", "3"]
    importfile.write(lstRow[0] + "," + lstRow[1] + "\n")
    importfile.close()
    importfile = open(objFile, "r")
    skipfileline = importfile.readlines()[1:]
    for row in skipfileline:
        lstRow = row.split(",")
        dicRow = {"Task": lstRow[0], "Priority": lstRow[1].strip()}
        lstTable.append(dicRow)
    importfile.close()

# -- Input/Output -- #
```

```
1 {'Task': 'Wash Windows', 'Priority': '2'}
2 {'Task': 'Clean Dishes', 'Priority': '3'}
3 {'Task': 'Pressure Wash House', 'Priority': '4'}
Enter an ID to Remove:0

    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

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

{'Task': 'Wash Windows', 'Priority': '2'}
{'Task': 'Clean Dishes', 'Priority': '3'}
{'Task': 'Pressure Wash House', 'Priority': '4'}
Saving above list to file

    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

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

Goodbye
PS C:\_PythonClass\Module05\Module05 - Lists and Dictionaries\Assignment> python .\Assigment05_Starter.py

    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

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

This the current To Do List:
{'Task': 'Wash Windows', 'Priority': '2'}
{'Task': 'Clean Dishes', 'Priority': '3'}
{'Task': 'Pressure Wash House', 'Priority': '4'}

    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

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

Goodbye
PS C:\_PythonClass\Module05\Module05 - Lists and Dictionaries\Assignment>
```

```
Task,Priority
Wash Windows,2
Clean Dishes,3
Pressure Wash House,4
```

Figure 5 – Proof of Execution