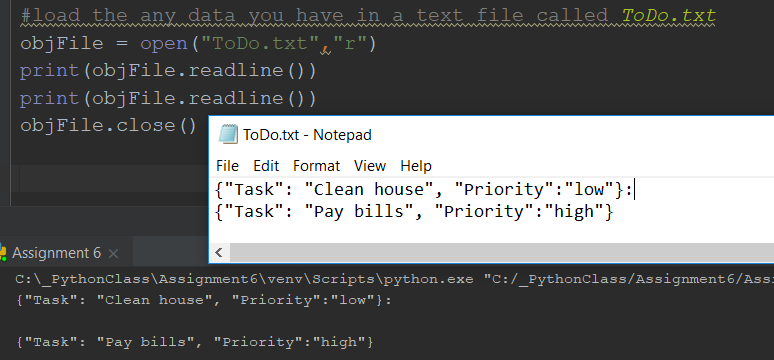
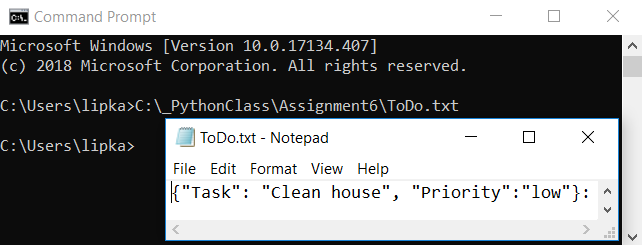
Jessica Lipka  
Nov 19th 2018  
IT FDN 100 B  
Assignment05

Functions

Introduction: In this assignment I am using the ToDo list from the previous assignment and place that code into function and a class.

**Step 1:** load the any data you have in a text file called ToDo.txt. I do this by using read.

  
***(figure1a example of using read to pull data from a text file)***

Copying the path will also run the ToDo file thru cmd shell.  
  
***(figure1b running in cmd shell)***

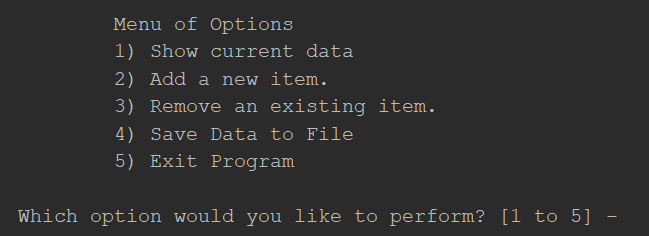
Define the functions – The functions need to be defined in order to be used later in the program.

**Step 2:** Display a menu of choices to the user by defining the function menuoptions   
Also begins the while loop to keep the program open until it reaches the end.

#Display a menu of choices to the user  
while(True):  
 def menuoptions():  
 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  
 """)

***(figure2a defining function menuoptions)***  
  
Then I call the function I defined and add input to be able to choose an option.

menuoptions()  
strChoice = str(input("Which option would you like to perform? [1 to 5] - "))

***(figure2b calling the defined functions)***  
The above code returns:  
  
***(figure2c menu displayed to user)***

**Step 3:** Display all todo items to user by defining the function ToDo.

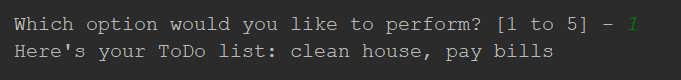
# Display all *todo items to user* def ToDo(lstTable):  
 print("Here's your ToDo list: " "clean house, " "pay bills ")

***(figure 3a defining function todo)***  
  
Then I call the function if defined, use an if clause to make sure the correct option displays the correct output, and use continue to keep the program open.

if (strChoice == '1'):  
 ToDo(lstTable)  
 continue

***(figure3b calling the defined function)***

The above code returns

  
***(figure3c displayed to user)***

**Step 4:** Add a new item to the list/Table  
This is where everything goes downhill. I couldn’t figure out how to get my add function to work with the dictionary list. You can see I commented it out. It runs without the add, but that’s not how it’s supposed to be done.

# Add a new item to the list/table  
 def add():  
 input("What to add:? ")

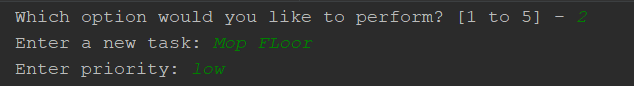
***(figure4a trying to add user input to a dictionary list through defining a function)***

if (strChoice == '2'):  
 #add()  
 strT = input("Enter a new task: ")  
 strP = input("Enter priority: ")  
 dicNewRow = {"Task": strT, "Priority": strP}  
 lstTable.append(dicNewRow)  
 print(lstTable)  
 continue

***(figure4b trying to add user input to a dictionary list through defining a function)***

I even brought over the dictionary list. I tried to define dicRow0 as a function, but it returned an error. I tried def additem(dicRow0), but that also returned an error because of the str.

dicRow0 = {"Task": "Clean house", "Priority":"low"}  
dicRow1 = {"Task": "Pay bills", "Priority":"high"}  
lstTable = [dicRow0,dicRow1]

***(figure4c trying to add user input to a dictionary list through defining a function)***  
  
  
  
***(figure4d trying to add user input to a dictionary list through defining a function)***

**Step 5:** Remove a new item to the list/Table. Again with this part I can get it to run, but it’s not as it is supposed to be done. I defined a function, albeit an incorrect one, and since I couldn’t figure out how to append the dictionary to add a list, then I surely can’t remove a list.

# Remove an item from the list/table  
 def remove():  
 input("Choose task to delete [1 or 2] - ")

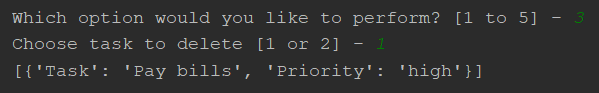
***(figure5a trying to define a function)***

The second part of the if/elif is commented out because it doesn’t work because nothing else is working.

if (strChoice == '3'):  
 remove()  
 if str(input == '1'):  
 lstTable.remove(dicRow0)  
 print(lstTable)  
 #elif str(input == '2'):  
 # lstTable.remove(dicRow1)  
 # print(lstTable)  
 continue

***(figure5b trying to use an if/elif clause)***

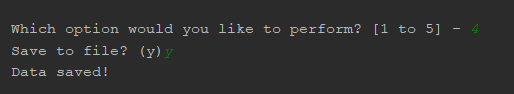
It’ll run and selecting option 1 will remove list 1. But selecting option 2 does nothing.

  
***(figure5c user input and output)***

**Step 6** Save tasks to the ToDo.txt file. Again, trying to define a function for this. Everything runs and displays correctly but I doubt its really saving anything.

# Save tasks to the *todo.txt file* def question():  
 input('Save to file? (y)')

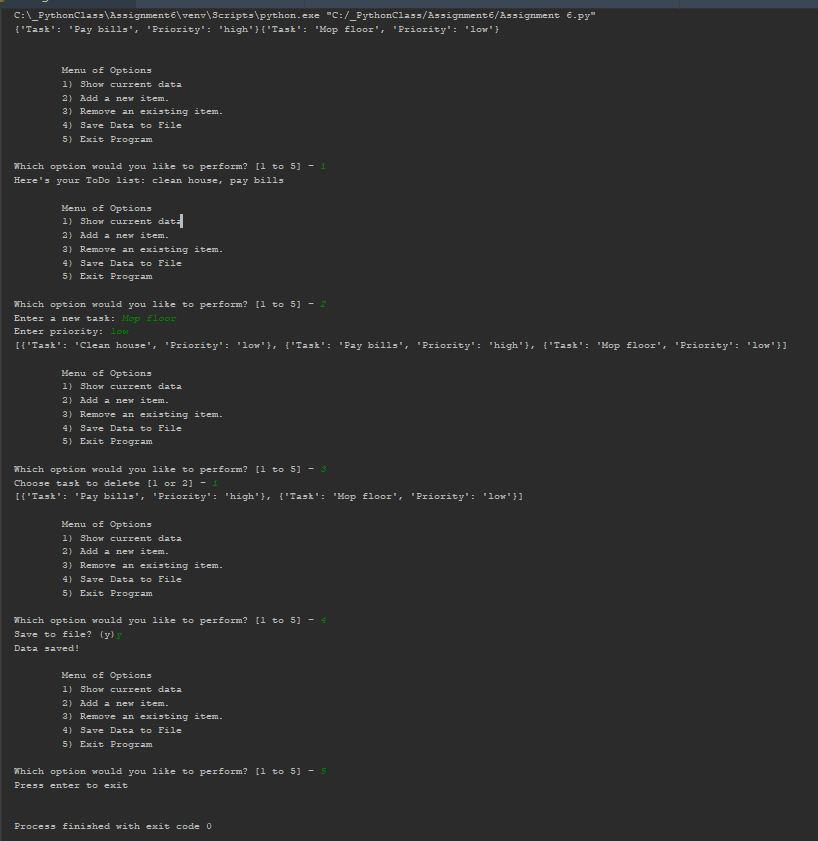
if (strChoice == '4'):  
 question()  
 objFile = open('Todo.txt', 'w')  
 for row in lstTable:  
 objFile.write(str(row))  
 print("Data saved!")  
 continue



**Step 7** Exit program. This, I think is correct.

# Exit program  
 def exit():  
 print(input("Press enter to exit"))

if (strChoice == '5'):  
 exit()  
 break



**Summary:**I did not understand this module at all. None of the videos or other resources explained this concept in a way that made sense to me.