Julian Herndon

November 17, 2020

Foundations of Programing: Python

Assignment 05

https://github.com/Raphidophyte/IntroToProg-Python

Python “ToDo List.py”

Preamble of Frustration:

Learning to code seems somewhat like learning to write in a new language. First there is a basic need to know the meaning of the words and how to spell them, then one needs to understand the syntax of how words in the new language fit together to convey a message and finally sentence structure and conjugation of the verb tenses along with proper paragraph construction are key in order for the message to be clear to the reader. With coding, one must learn the meaning of the various functions or commands, how they are written and what they mean or do. Then you need to learn how to put those commands, functions and sub-code together in the proper syntax so that intent can be conveyed, kind of like pseudocode and finally there is a requirement that the statements be constructed correctly and matched with other statements in such a way that the code will produce the final desired result. This is huge challenge so far. Learning any single aspect described above and being able to write this out in a console application from scratch is a huge challenge. Using examples as a guide help, but unlike human language, where rudimentary knowledge is sufficient to get by, in coding there can be no mistakes.

Introduction:

While Assignment05 “ToDoList.py” makes use of skills learned in the previous four assignments, but this project is more complex. The script is intended to create a basic “ToDo List” with a setoff tasks and an assigned priority. This Python script first loads data from a .txt file to memory and then presents the user with a five-point menu to choose an action from, including showing what is in the .txt file already, entering new tasks, the ability to delete an existing task and finally saving and exiting. The script initially creates a new text document and then appends new data to it. The functional portions of code were add into starter code provided Randal Root. This script was written using PyCharm.

Asignment05 questions and answers:

1. What is the difference between a List and a Dictionary?

Answer: A list and a Dictionary are similar, but in a list, the objects of elements are accessed with an index (numeric enclosed in square brackets []) whereas in a Dictionary the elements are accessed using a key (character enclosed in braces, {}).

1. What is the between an Index and a Key?

Answer: An index is number in brackets (i.e. [0], [1], etc.), and a key is a character in braces with quotes (i.e. {“id”}, {“name”}, {“email”}). Both of these are subscripts.

1. How do you read data from a file into a List?

Answer:

objFile = open(strFile, “r”) # where strFile was previously defined by the user as a .txt file

for row in objFile:

lstRow = row.split(“,”) # where the .split is going to separate the data items in the row using a coma (comma delimited)

print(lstRow)

print(lstRow[0] + lstRow[1] + lstRow [2].strip()) # the square brackets slice the items in the list and show them individually & the .strip is removing any function at the end of the last data item, like say a carriage return (\n).

objFile.close()

1. How do you read data from a file into a Dictionary?

Answer:

# Declare variables

strFile = “MyData.txt”

objFile = None

# Create the table

dicRow = {}

lstTable[]

# Process the data

dicRow = {“id”: “1”. “name: “Bob Smith”} # Notice the use and placement of the colon (:).

lstTable.append(dicRow)

for objRow in lstTable:

print(objRow)

# Get User Input

strID = input(“Enter and ID: “)

strName = input(“Enter an Email: “)

dicRow = {“id”:stride,”name”:strName)

lstTable.append(dicRow)

for objRow in lstTable

print(objRow)

1. What is the programming pattern called “Separations of Concerns?”

Answer: Separation of Concerns (SoC) is a principle of program design of splitting programs into sections that address separate concerns. This may help make programs more efficient in terms of memory use or allow separate components to be used in other programs or interact on their own with other programs running on a system. For instance separating a program into data, processing and presentation.

1. How would you use a function to organize your code?

Answer: By organizing code into sections that you can control. A function will allow a programmer to group a set of statements and define them by giving them a name and then the function can be “called” later in the script when needed. The function is loaded into memory and waits to be called to do its job. By this means, once a function is defined, it can be called upon in the script more than once without having to re-type it each time it is needed. The script will run the function when it is called and then move on to the next statement. This improves flow control. The function can be called with the print command.

1. Why is a script template useful?

Answer: A script template is like a header for your script that you can save and reuse by adding it to all your scripts. It provides consistency to your programs, simplifies professional interactions and helps identify your code quickly.

1. Why is error handling using Try-Except recommend?

Answer: Error handling (Try Except) allows the programmer to manage errors made by users. It provides user friendly error messages or allows the python to move to another set of statements if thee is an error it cannot handle is one set of statements. “You can trap errors in your programs using a try-except construct.” (Root, Mod05 video, 2019).

1. What is GitHub, and why is it used?

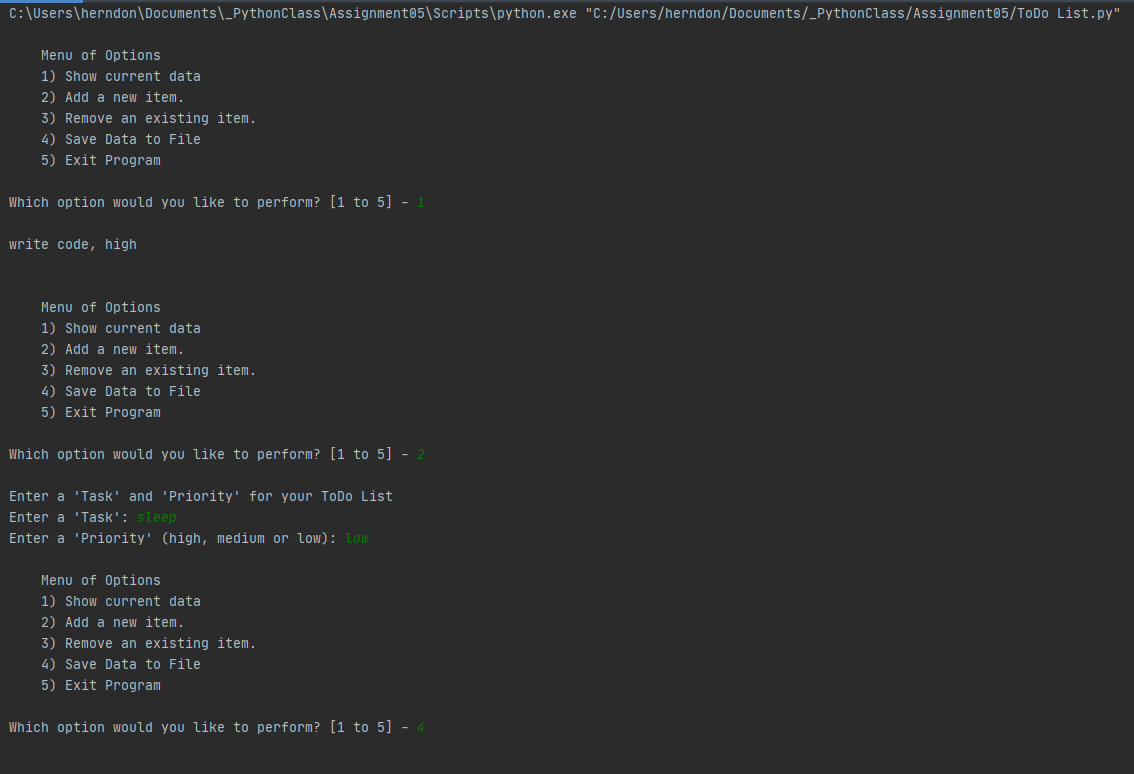
Answer: GitHub is a website (GitHub.com) where code can be stored using the source control software Git. This allows a programmer to make a backup of their code files and also keep track of its revision history in a central repository when collaborating with other programmers.

The script:

It is possible that my least favorite message to see on Python is “string index out of range”, closely followed by “AttributeError”. In Assignemnt05, python script has become a bit of an educated guess game to me. While the code we have worked on up until this point has been relatively simple and straightforward, at this point there are now sufficient variables to make it difficult to know why something is not working correctly!

A detailed list of steps attempted and found not to work was not generated, but suffice it to say that there were many combinations of command, functions and code statements and syntax attempted. The script is not fully functional. The sections that work well borrow heavily from script found in lab 5-2 and Assignment04 (HomeInventory). Step 1 (load data from the text file to memory) may or may not work. It does load and run without crashing the program, which was not always the case! Step 2 (show current data) works, but this section requires a .txt file to already be standing by with some data in it. Step 3 (remove existing data) does not work. This was the most frustrating section. I tried to use a “Try-Except” construct to move around that section of the code, but was not able to get it to function correctly, so, for now that Step 3 is commented out and it simply prints “Task not in file”. Section 4 (save data) works somewhat, but currently it saves replica rows of data in the .txt file. Section 5 (exit) does work. The script will function only in PyCharm, a portion of which can be seen in Figure 1, and will save data to the “ToDoList.txt” file (Figure 2). When started at the CMD prompt, I get a message of “no such file or directory” (Figure 3) for the “ToDoList.txt” file, so the script does not run.

The script was developed through the now usual iterative process using the Assignemnt05.pdf (Root undated), the Mod5PythonProgrammingNotes (Root undated) and Mod05 Course Video (Root 2019) as primary guides and an assortment of internet searches in an attempt to discover examples of proper syntax for “while loop”. Below is the basic working code being submitted (figure 1).

Figure 1: “ToDoList.py” running in PyCharm.

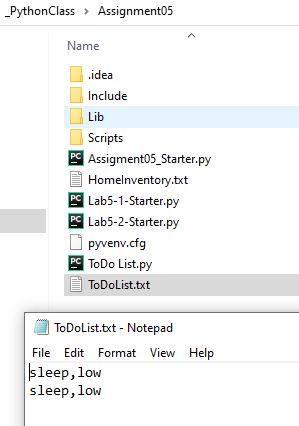


Figure 2: “ToDoList.py” data saved to .txt file.

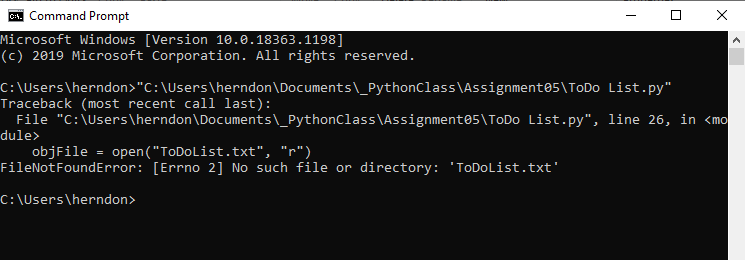


Figure 3: Error when trying to run “ToDoList.py” at the CMD prompt.

Summary:

Overall, this assignment failed to achieve a satisfactory, fully functioning script. I anticipate that many of the errors are simple syntax mistakes compounded by a lack of clear understanding of how to create a functioning dictionary with data stored in a table format in a .txt file. On the bright side, the github.com link should work!

References:

Python.org, (2020). https://docs.python.org/3/library/functions.html

Root, R. Intro to Programming (Python) Assignment 05. Undated.

https://canvas.uw.edu/courses/1417585/files/67261869?module\_item\_id=11076718

Root, R. Mod5 Python Programing Notes (undated)

https://canvas.uw.edu/courses/1417585/files/67261864?module\_item\_id=11076713

Root, R. Mod05 Course Video. 10/28/2019.

https://www.youtube.com/watch?v=P5wOsnPjn6Y&feature=youtu.be