Kevin Wong

November 13, 2019

Foundations of Programming: Python

University of Washington

Professor: Randal Root

Assignment 06

**Introduction**

For assignment 6, we are to modify an existing script that manages a "ToDo list." The code only uses a few functions, and our job is to add more functions to organize the code. The function of program is similar to what was given in assignment 5.

**Assignment**

We are provided a starting template script to modify as our own to complete this assignment. This new script manages a “ToDo list” text file. The text file contains two columns of data, “Task” and “Priority”. The user has option to view, add, and remove items from the text file. The script is very simialr to what our Assignment 5, but in this case, we are asked to add more functions and organize the code.

I have organized the pseudocode based on the steps to run the programs as provided in a existing script. I’ve also added the existing and new functions I created to the psuedocode. The old and new functions are colored in blue and green respectively.

**Step 1 - Load data from ToDoFile.txt**

* **FileProcessor.ReadFileDataToList(strFileName, lstTable)**
  + Load data from ToDoFile.txt into a table

**Loops through Steps 2 and 3**

**Step 2 – Display a menu of choices to the user**

* **IO. OutputMenuItems()**
  + Displays menu option
* **IO. InputMenuChoice()**
  + Get user input for menu option

**Step 3 - Process user's menu choice [1-6]**

**Step 3.1**

* If Choice == 1 (display current table)
  + **IO.ShowCurrentItemsInList(lstTable)**
    - Display current table

**Step 3.2**

* If Choice == 2 (Add a new item to the list/Table)
  + **IO.AddDataToList(lstTable)**
    - Step 3.2a - Ask user for task and priority
    - Step 3.2b - Append table with the new task and priority

**Step 3.3**

* If choice == 3 (Remove a new item to the list/Table)
  + **IO.RemoveDataFromList(lstTable)**
    - Step 3.3a - Ask user which item to remove
    - Step 3.3b – Remove item from table
    - Step 3.3c – Display update to user whether task was removed or not
  + **IO.ShowCurrentItemsInList(lstTable)**
    - Step 3.3d – Display current table

**Step 3.4**

* If choice == 4 (Save tasks to the ToDoFile.txt file)
  + **IO.ShowCurrentItemsInList(lstTable)**
    - Step 3.4a – Display current table
  + **FileProcessor.WriteListDataToFile(strFileName, lstTable)**
    - Step 3.4b – Ask yes or no to save data

**Step 3.5**

(Reload data from the ToDoFile.txt file (clears the current data from the list/table)

* If choice == 5

**FileProcessor.ReloadDataFromFile(strFileName, lstTable)**

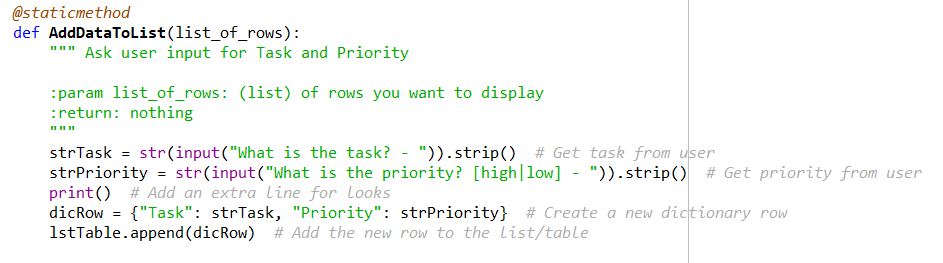
**Step 3.6**

* If choice == 6 (Exits the program)
  + **Exit Program**

**New Functions**

**IO.AddDataToList(lstTable)**

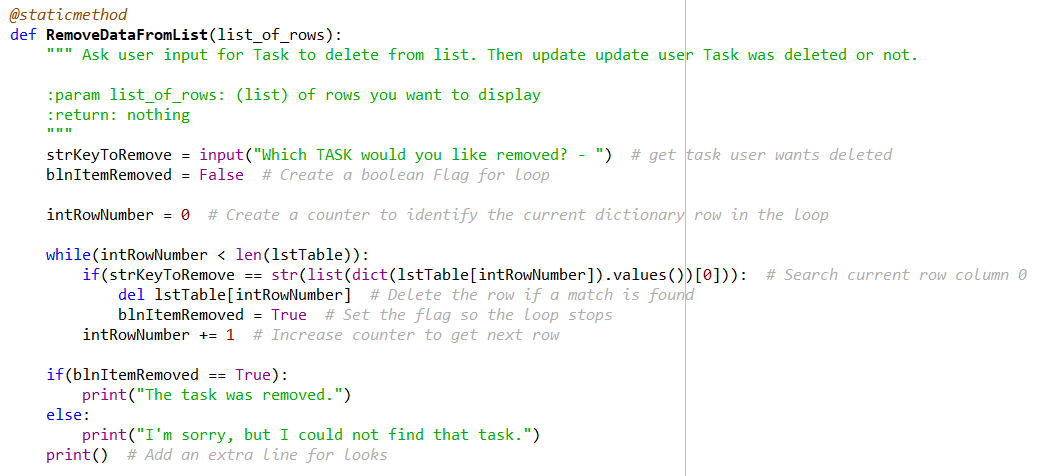
The new function **AddDataToList()** is used to ask for user input to new Task and Priority to add to the current list. The code is ahown in Figure 1.



**Figure 1. Screenshot of new function AddDataToList().**

**IO.RemoveDataFromList(lstTable)**

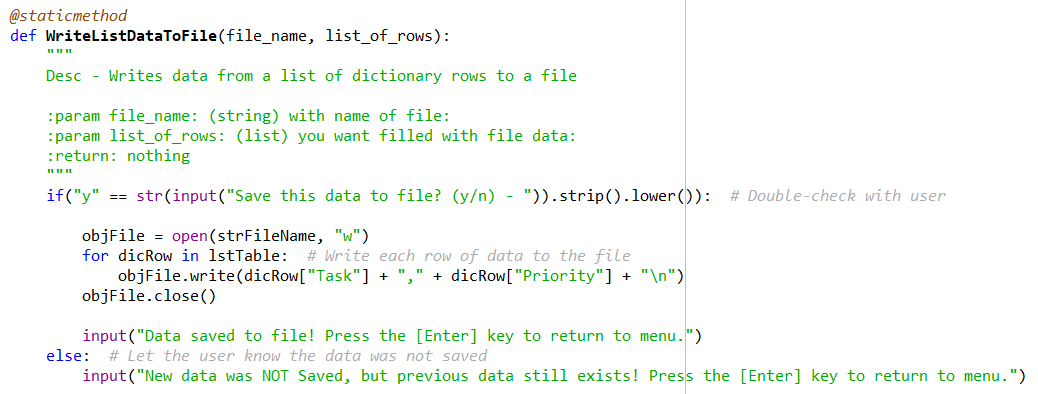
The next new function is **RemoveDataFromList()** which is used to remove data from the text file. The function will first ask the user which task he/she would like to remove. If the user inputs a task that is not currently in the list, the progrma will respond that it could not find the task. If not, it will remove the task and update the user as such. The code for the function is shown in Figure 2.



**Figure 2. Screenshot of the new function RemoveDataFromList().**

**FileProcessor.WriteListDataToFile(strFileName, lstTable)**

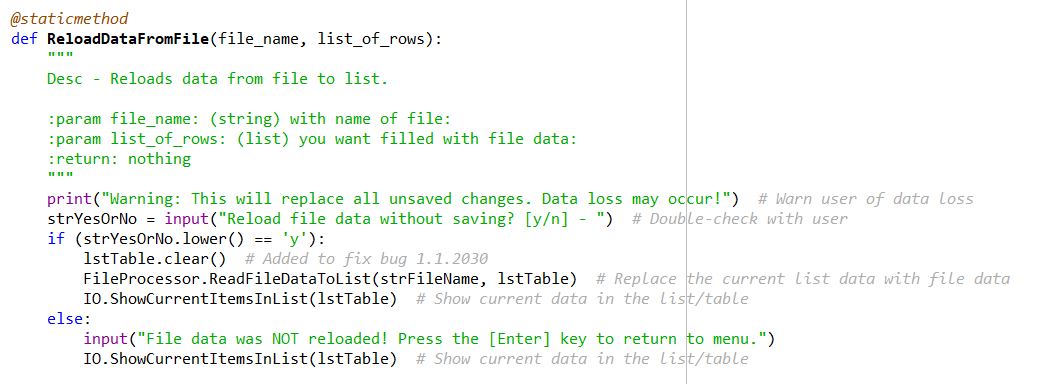
The 3rd new function is called **WriteListDataToFile()** and used to saved the list of tabled items to the text file, “ToDoFile.txt”. The program will firstly ask the user a yes-or-no question if he/she would like to save the data to the text file. If the user enters “y”, then the data is saved as the program will response that the file have been saved. If the user inputs “n”, then the program will not save the data and displays that the data have not been saved. The code is show in Figure 3.



**Figure 3. Screenshot of the new function WriteListDataToFile().**

**FileProcessor.ReloadDataFromFile(strFileName, lstTable)**

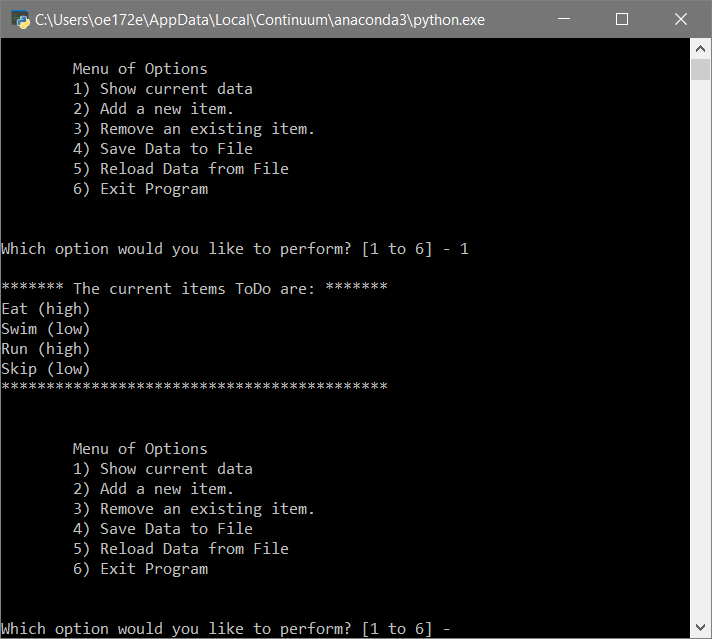
The last new function is called **ReloadDataFromFile()** and use to reload the data from the text file. The function will first inform the user that this procedure will replace all unsaved changes before proceeding to ask yes or no to reload the data. The code is shown in Figure 4.



**Figure 4. Screenshot of new funtion ReloadDataFromFile().**

**Testing the Script**

The script is then tested to ensure that all the code work as expected, as shown in Figure 5.



**Figure 5. Screenshot of the script working in the console.**

**GitHub**

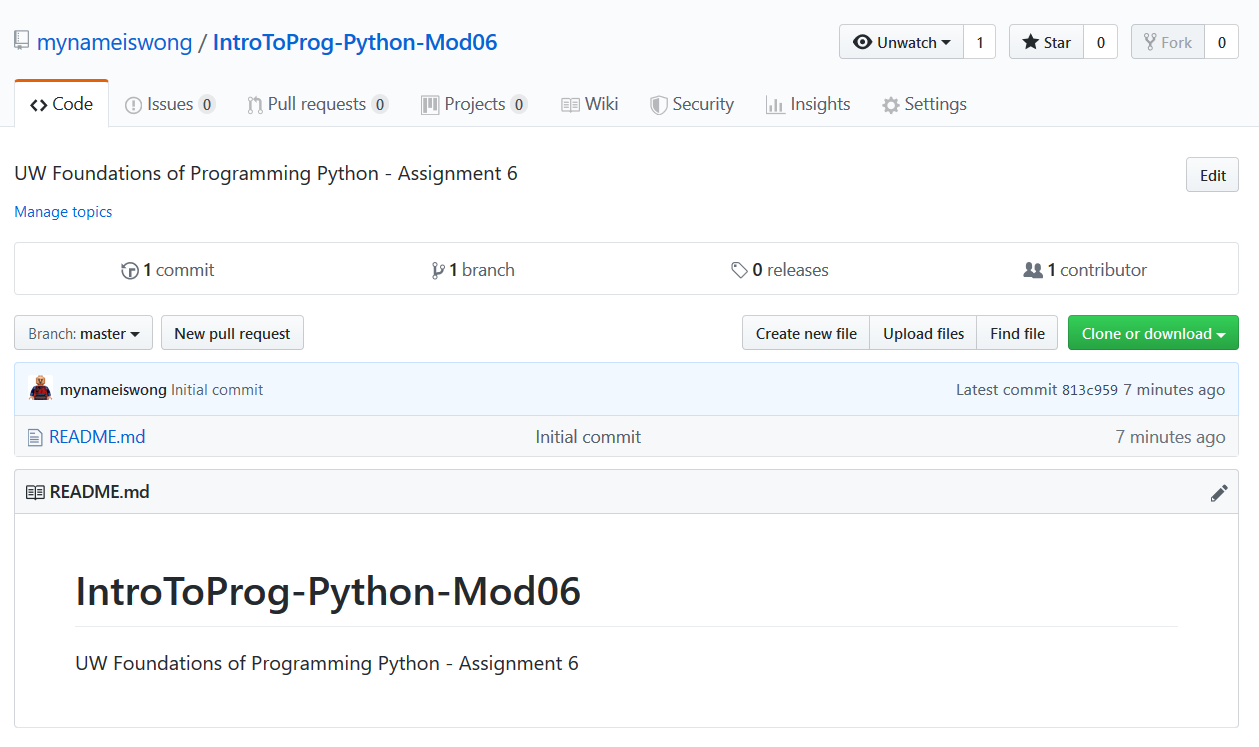
We are then to post our assignment files into a public GitHub repository so that others may review it. We also learn how to add a GitHub webpage to our respository.

Repository name: **IntroToProg-Python-Mod06**

Github repository: <https://github.com/mynameiswong/IntroToProg-Python-Mod06>

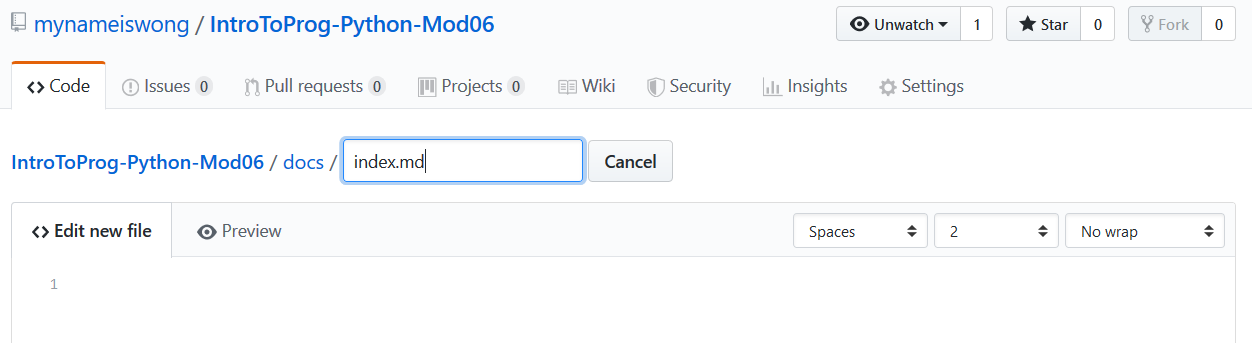
Github webpage: <https://mynameiswong.github.io/IntroToProg-Python-Mod06/>

After creating our new repository for assignment 6, we the then learn to create a new folder by clicking on to “Create new file” button as shown in Figure 6.



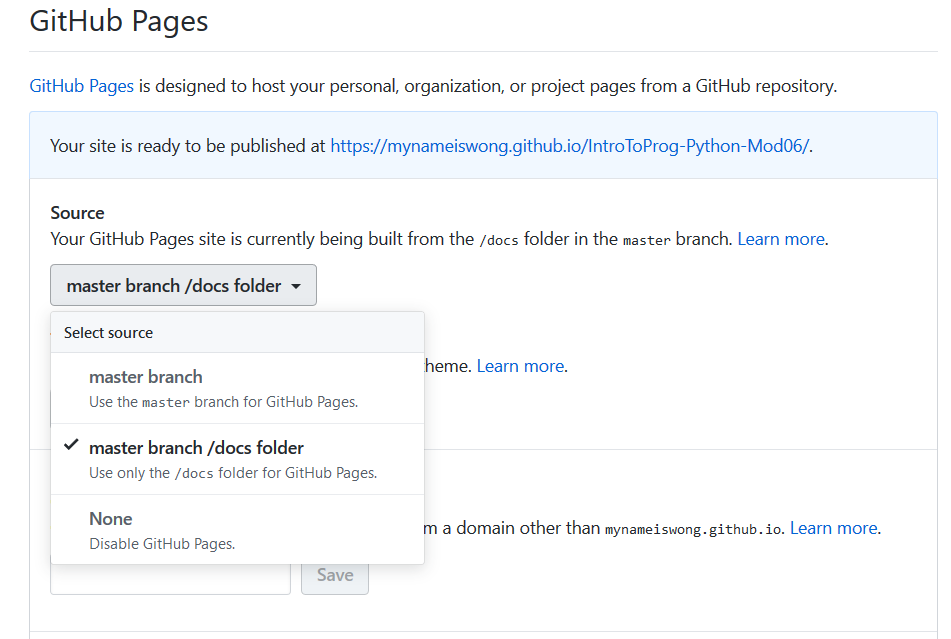
**Figure 6. Screenshot of the GitHub respository showing the button *Create new file*.**

We then name our new folder as docs/index.md, which as a main webpage for our GitHub folder, as shown in Figure 7.



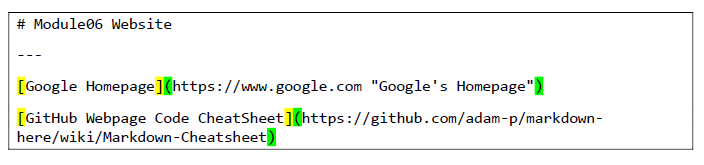
**Figure 7. Screenshot of how to create the main GitHub folder.**

After that, we will click on **Settings** and change the **Source** to **master branch/docs folder** as shown in Figure 8.



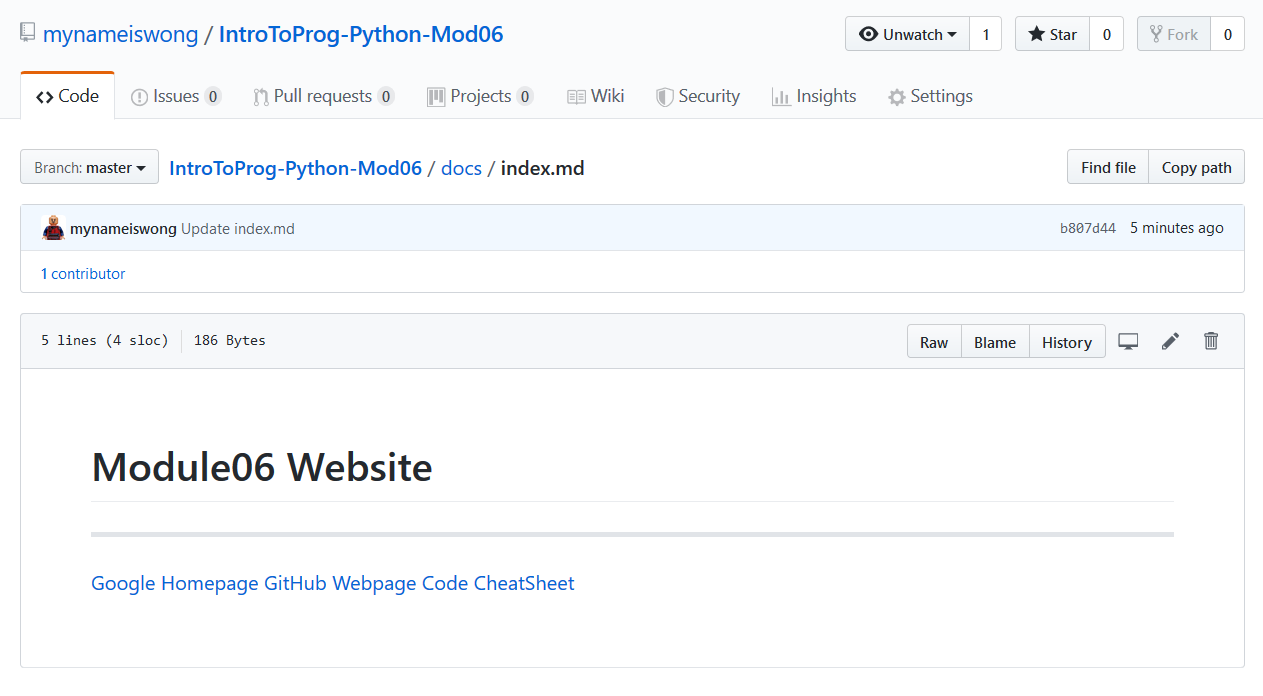
**Figure 8. Screenshot of setting page to change the Source.**

We then copy and paste the following text, as shown in Figure 9, from the Module 6 Programming Notes to the GitHub index file.



**Figure 9. Screenshot of the text from the programming notes to add to the GitHub webpage.**

The resulting GitHub webpage is now shown in Figure 10.

**Figure 10. Screenshot of the GitHub webpage.**

**Readings**

1. **Chapter 6 ofPython Programming for the Absolute Beginner(Third Edition) by Michael Dawson.**

## **Module 6 PDF - Mod5PythonProgrammingNotes**

## **Functions Tutorial**

## <http://www.learnpython.org/en/Functions>

## **Markdown Cheatsheet**

## <https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet>

# Writing on GitHub

# <https://help.github.com/en/github/writing-on-github>

**Videos**

# Intro to Python Mod06 by Randal Root

# <https://youtu.be/jiXmXhwgHp8>

# Let's Learn Python - Basics #6 of 8 – Functions

# <https://youtu.be/qO4ZN5uZSVg>