Patrick Chuoy

August 16, 2023

Foundations of Programming: Python

Assignment 06

https://github.com/trickyUW/IntroToProg-Python-Mod06

To Do List (Using Functions)

Introduction

Module 06 was all about functions. I learned about functions, parameters, return, global vs. local variables, and classes. I also learned how to debug in PyCharm and create a GitHub website. Assignment 06 is the To Do List script from Assignment 05, except the code is more organized by using functions. A starter file was given, and I added code to make the script work.

Understanding Functions

Functions are a way of grouping one or more statements together. The function can then be called later in different parts of the script. Big scripts can be split up and organized better when using functions. Once you understand the order in which code executes and where to look when functions are called, scripts become more readable. Functions can split statements into smaller chunks and also be reused without copying and pasting a lot. The main body of the script becomes more digestible and closer to pseudo code, making it easier to understand.

Adding Code

A starter Python file was given for Assignment 06. It was already organized and had a good amount of code. The script was almost the same as Assignment 05 but implemented differently. There were spots in the script that were commented with, "# To Do," where I had to add code. Most of the spots were in the functions, such as add_data_to_list or input_new_task_and_priority. The functions were grouped into two classes, Processor, and IO. Then in the main body of the script, the functions were called when needed. All I had to do was add code from Assignment 05 to the functions in Assignment 06. For example, in Assignment 05 step 4 (Adding a new item to the list), there was code I needed for the input_new_task_and_priority and the add_data_to_list function.

Error Handling

When I first tried to test the script, there was no ToDoList.txt file, so I got an error right away. To fix this, I needed to create the file and populate it with data. To have better error handling, I wanted to add a Try-Except block since I didn't in Assignment 05. In the read_data_from_file function, I put the code for opening the file and adding rows to the list in a Try-Except block.

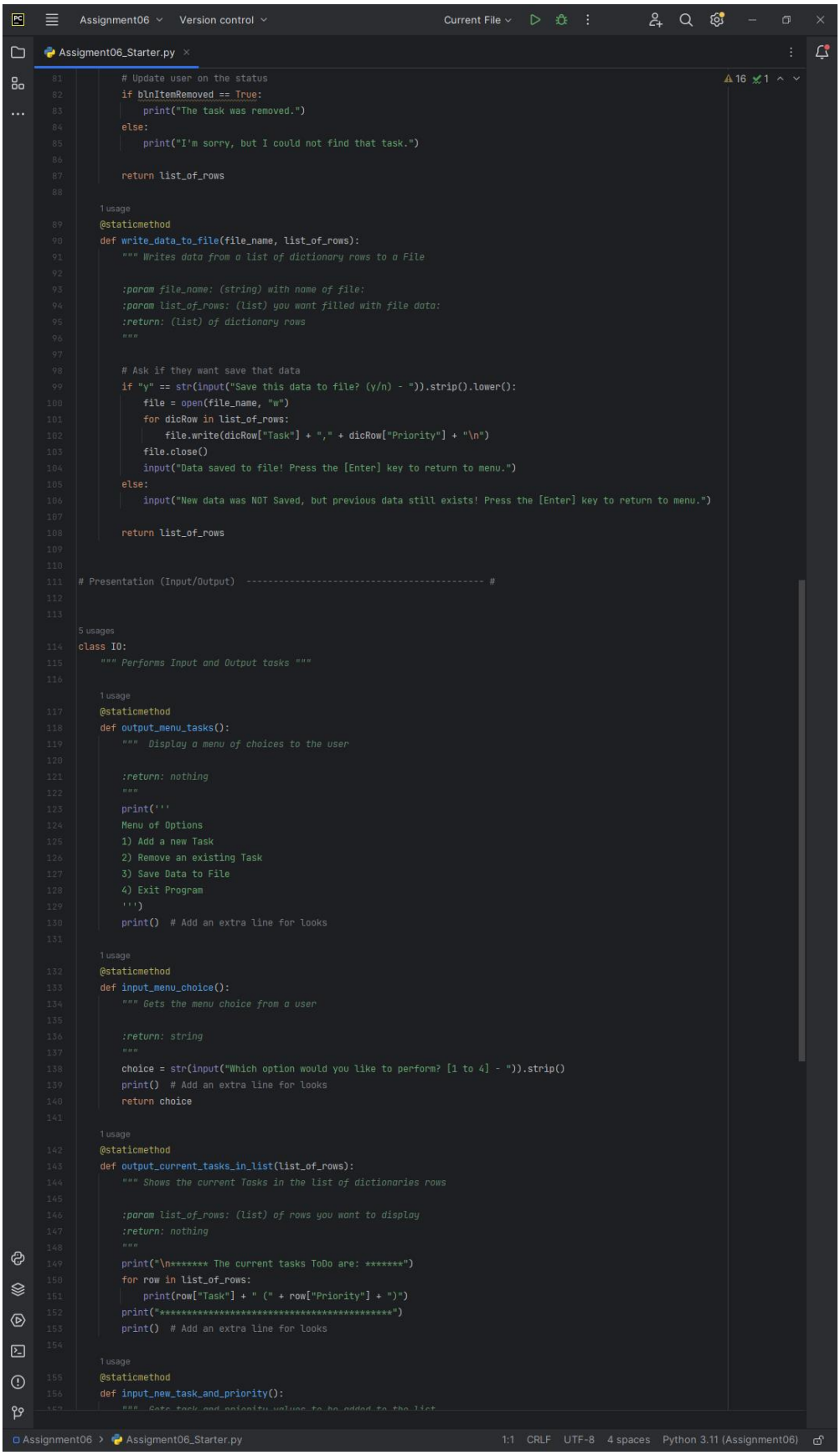
Practice

Modifying and adding to existing code is good practice for the real world. It's a good way to learn and can feel like a puzzle to me. Concepts like functions and classes are important to programming. When the code is already there for me, it can be hard to remember later or when I need to create my own script from scratch. I will try to look over the code that was already given in Assignment 06 to better understand how and why things were done the way they were.

Summary

Assignment 06's To Do List script was like a puzzle to me. The script had a similar output to Assignment 05 but organized the code better by using classes and functions. Using what I learned from Module 06, I added code to get the script to work.


Assignment06
Version control
Current File
Assigment06_Starter.py
# -- #
# Title: Assignment 06
# Description: Working with functions in a class,
#   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
# PChuoy,8.15.23,Modified code to complete assignment 06
# -- #
# Data --- #
# Declare variables and constants
file_name_str = "ToDoFile.txt" # The name of the data file
file_obj = None # An object that represents a file
row_dic = {} # A row of data separated into elements of a dictionary {Task,Priority}
table_lst = [] # A list that acts as a 'table' of rows
choice_str = "" # Captures the user option selection
# Processing -- #
4 usages
class Processor:
""" Performs Processing tasks """
1 usage
@staticmethod
def read_data_from_file(file_name, list_of_rows):
""" Reads data from a file into a list of dictionary rows
:param file_name: (string) with name of file:
:param list_of_rows: (list) you want filled with file data:
:return: (list) of dictionary rows
"""
# Try-Except block for when ToDoFile.txt does not exist
try:
list_of_rows.clear() # clear current data
file = open(file_name, "r")
for line in file:
task, priority = line.split(",")
row = {"Task": task.strip(), "Priority": priority.strip()}
list_of_rows.append(row)
file.close()
except:
print("An error occurred when trying to open file.")
return list_of_rows
1 usage
@staticmethod
def add_data_to_list(task, priority, list_of_rows):
""" Adds data to a list of dictionary rows
:param task: (string) with name of task:
:param priority: (string) with name of priority:
:param list_of_rows: (list) you want to add more data to:
:return: (list) of dictionary rows
"""
row = {"Task": str(task).strip(), "Priority": str(priority).strip()}
# Add dictionary row of task, priority parameters to list
list_of_rows.append(row)
return list_of_rows
1 usage
@staticmethod
def remove_data_from_list(task, list_of_rows):
""" Removes data from a list of dictionary rows
:param task: (string) with name of task:
:param list_of_rows: (list) you want filled with file data:
:return: (list) of dictionary rows
"""
blnItemRemoved = False # Use this to verify that the data was found and removed
for row in list_of_rows:
list_task, list_priority = dict(row).values()
# If task parameter was found in list, remove row
if list_task.lower() == task.lower():
list_of_rows.remove(row)
blnItemRemoved = True
Assignment06 > Assigment06_Starter.py
1:1 CRLF UTF-8 4 spaces Python 3.11 (Assignment06)


Assignment06
Version control
Current File
Assigment06_Starter.py
16 1
# Update user on the status
if blnItemRemoved == True:
print("The task was removed.")
else:
print("I'm sorry, but I could not find that task.")
return list_of_rows
1 usage
@staticmethod
def write_data_to_file(file_name, list_of_rows):
""" Writes data from a list of dictionary rows to a File
:param file_name: (string) with name of file:
:param list_of_rows: (list) you want filled with file data:
:return: (list) of dictionary rows
"""
# Ask if they want save that data
if "y" == str(input("Save this data to file? (y/n) - ")).strip().lower():
file = open(file_name, "w")
for dicRow in list_of_rows:
file.write(dicRow["Task"] + "," + dicRow["Priority"] + "\n")
file.close()
input("Data saved to file! Press the [Enter] key to return to menu.")
else:
input("New data was NOT Saved, but previous data still exists! Press the [Enter] key to return to menu.")
return list_of_rows
Presentation (Input/Output) ----------
5 usages
class IO:
""" Performs Input and Output tasks """
1 usage
@staticmethod
def output_menu_tasks():
""" Display a menu of choices to the user
:return: nothing
"""
print('''
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Exit Program
''')
print() # Add an extra line for looks
1 usage
@staticmethod
def input_menu_choice():
""" Gets the menu choice from a user
:return: string
"""
choice = str(input("Which option would you like to perform? [1 to 4] - ")).strip()
print() # Add an extra line for looks
return choice
1 usage
@staticmethod
def output_current_tasks_in_list(list_of_rows):
""" Shows the current Tasks in the list of dictionaries rows
:param list_of_rows: (list) of rows you want to display
:return: nothing
"""
print("\n******* The current tasks ToDo are: *******")
for row in list_of_rows:
print(row["Task"] + " (" + row["Priority"] + ")")
print("***")
print() # Add an extra line for looks
1 usage
@staticmethod
def input_new_task_and_priority():
Assignment06
Assigment06_Starter.py
1:1 CRLF UTF-8 4 spaces Python 3.11 (Assignment06)

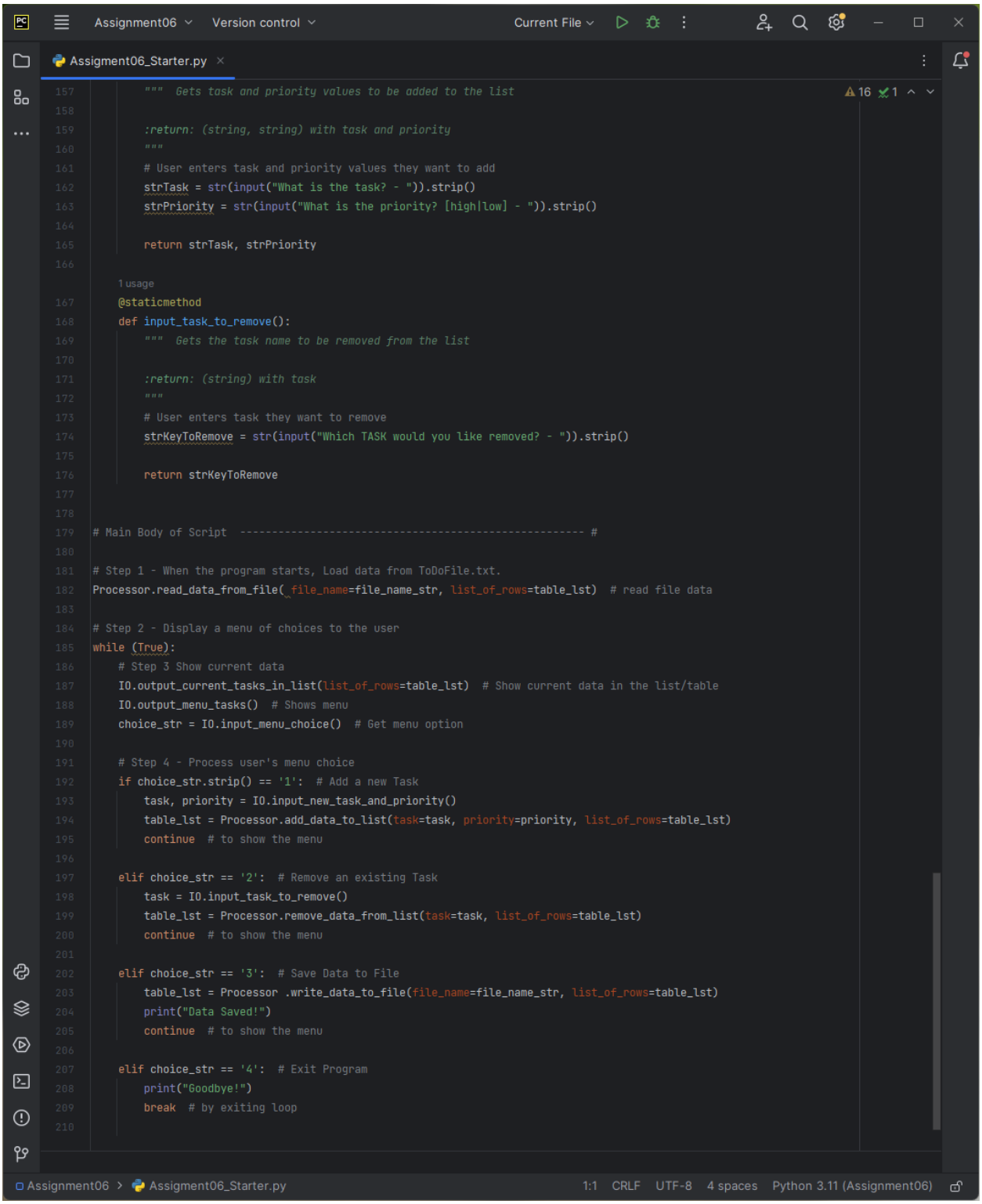
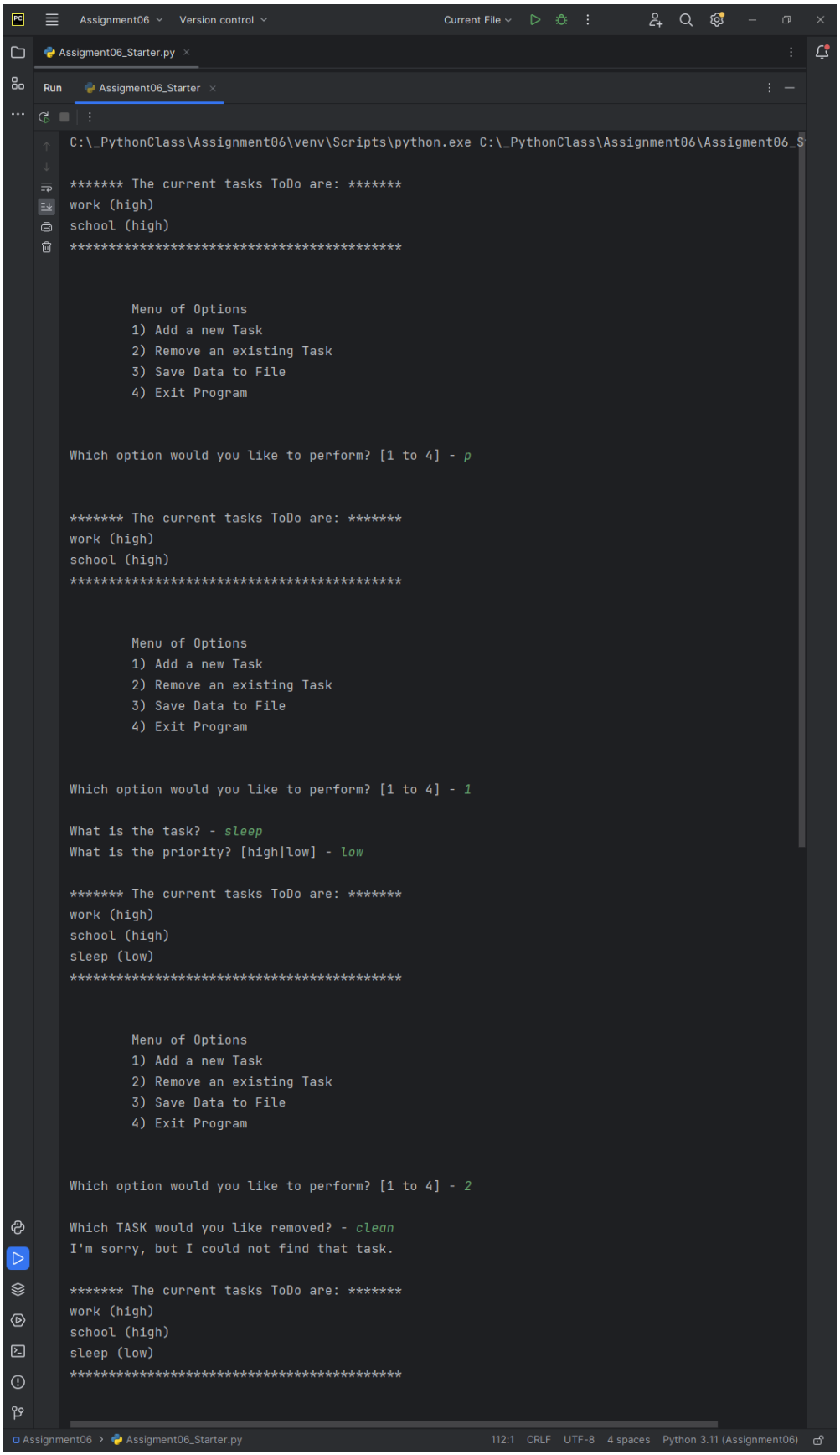


Figure 1: Script in PyCharm


Assignment06
Version control
Current File
Assigment06_Starter.py
Run
Assigment06_Starter
C:_PythonClass\Assignment06\venv\Scripts\python.exe C:_PythonClass\Assignment06\Assigment06_S
******* The current tasks ToDo are: *******
work (high)
school (high)
****
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Exit Program
Which option would you like to perform? [1 to 4] - p
******* The current tasks ToDo are: *******
work (high)
school (high)
****
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Exit Program
Which option would you like to perform? [1 to 4] - 1
What is the task? - sleep
What is the priority? [high|low] - low
******* The current tasks ToDo are: *******
work (high)
school (high)
sleep (low)
****
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Exit Program
Which option would you like to perform? [1 to 4] - 2
Which TASK would you like removed? - clean
I'm sorry, but I could not find that task.
******* The current tasks ToDo are: *******
work (high)
school (high)
sleep (low)
****
Assignment06 > Assigment06_Starter.py
112:1 CRLF UTF-8 4 spaces Python 3.11 (Assignment06)

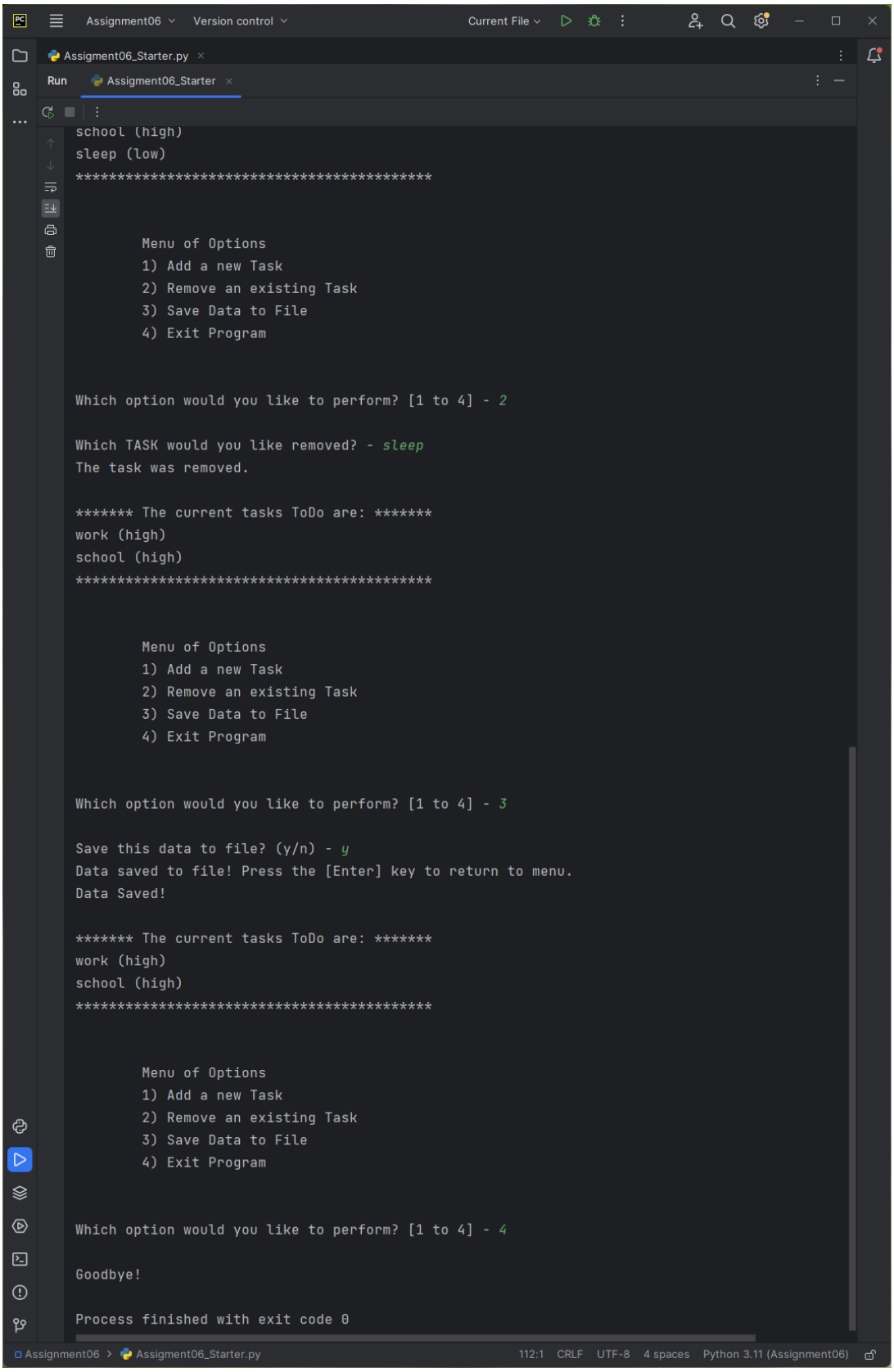


Figure 2: Script running in PyCharm

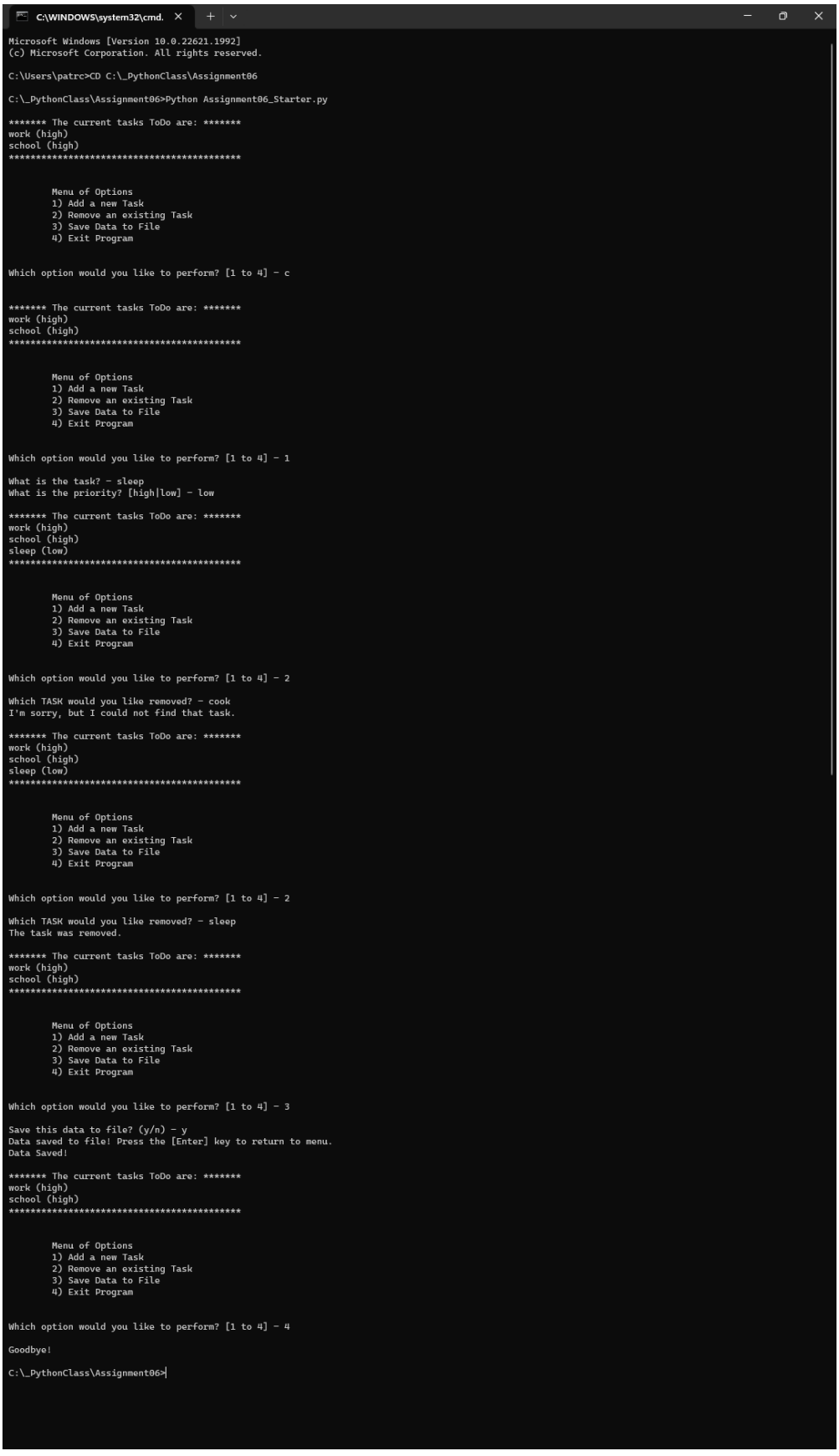

C:\WINDOWS\system32\cmd.
Microsoft Windows [Version 10.0.22621.1992]
(c) Microsoft Corporation. All rights reserved.
C:\Users\patrc>CD C:_PythonClass\Assignment06
C:_PythonClass\Assignment06>Python Assignment06_Starter.py
****** The current tasks ToDo are: *******
work (high)
school (high)
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Exit Program
Which option would you like to perform? [1 to 4] - c
****** The current tasks ToDo are: *******
work (high)
school (high)
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Exit Program
Which option would you like to perform? [1 to 4] - 1
What is the task? - sleep
What is the priority? [high|low] - low
****** The current tasks ToDo are: *******
work (high)
school (high)
sleep (low)
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Exit Program
Which option would you like to perform? [1 to 4] - 2
Which TASK would you like removed? - cook
I'm sorry, but I could not find that task.
****** The current tasks ToDo are: *******
work (high)
school (high)
sleep (low)
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Exit Program
Which option would you like to perform? [1 to 4] - 2
Which TASK would you like removed? - sleep
The task was removed.
****** The current tasks ToDo are: *******
work (high)
school (high)
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Exit Program
Which option would you like to perform? [1 to 4] - 3
Save this data to file? (y/n) - y
Data saved to file! Press the [Enter] key to return to menu.
Data Saved!
****** The current tasks ToDo are: *******
work (high)
school (high)
Menu of Options
1) Add a new Task
2) Remove an existing Task
3) Save Data to File
4) Exit Program
Which option would you like to perform? [1 to 4] - 4
Goodbye!
C:_PythonClass\Assignment06>

Figure 3: Script running in CMD

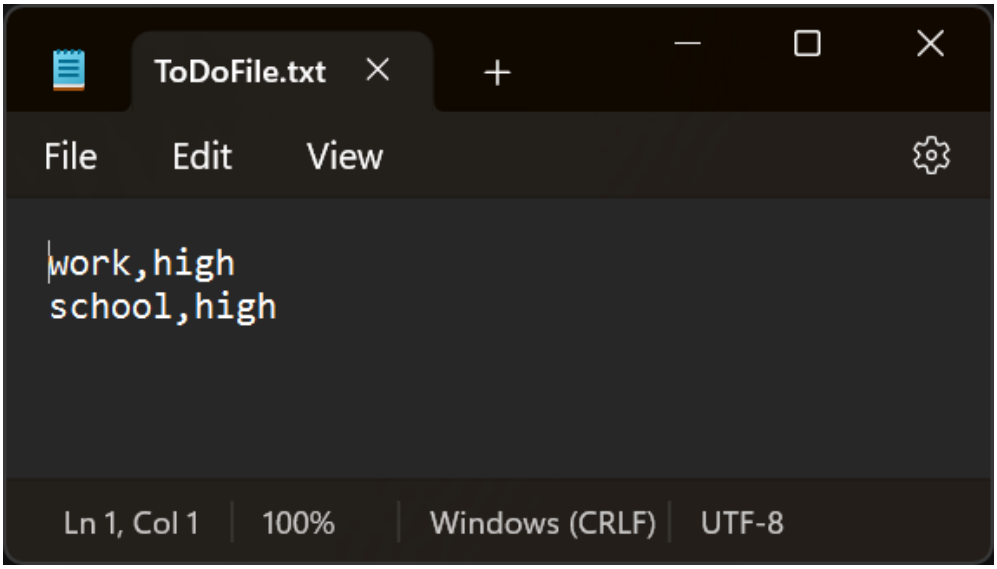


Figure 4: Text file