**Name:** Dusan Sretenovic

**Date:** February 21, 2024

**Course:** Foundations of Programming: Python

**Assignment 06 – Classes and Separation of Concerns**

**Introduction**

This document is going to describe the steps I took to perform this week’s assignment. Building off last week’s assignment, we will learn how to use classes, functions and the concept of separation of concerns.

**Defining Constants and Variables**

The first step was to define the constants and variables we will be using in this program.

A screenshot of a computer program

Description automatically generated

***Figure 1. Defining constants and variables***

**Creating Classes (File Processing)**

The first class we created was FileProcessor. This class contains two different functions (methods):

read\_data\_from\_file: This function opens the JSON file, reads the data into a list of dictionaries and closes the file,

write\_data\_to\_file: This function opens the JSON file in write mode, saves the list of dictionaries into the file and closes it.

Both functions also contain error handling, calling on the IO class which will be outlined later in the text. This is shown in Figure 2.

**A screenshot of a computer program

Description automatically generated**

***Figure 2. File Processor class and functions contained within.***

**Creating Classes (Input/Output)**

The next class we create is the IO (Input/Output) class. This class contains functions that manage user input and output. The functions are:

Output\_error\_messages: This function displays custom error messages to the user.

Output\_menu: This function displays a menu of choices to the user.

Input\_student\_data: This function asks the user to input the data for student’s first name, last name and course name. It also contains error handling for Value errors, such as entering non-alphabetical characters for the student’s name.

Output\_student\_course: This function displays the course information to the user, based on the input provided.

A screenshot of a computer program

Description automatically generated

***Figure 3.*** ***Class IO functions part 1.***

**A screen shot of a computer program

Description automatically generated**

***Figure 4. Class IO functions part 2.***

**Main Body of the Script**

After we’ve created the classes and defined the functions contained in the classes, we can write the main body of our script using those functions. We read the data from the JSON file using the FileProcessor class and the read\_data\_from\_file function within that class.

We open a while loop and output the menu using the Output\_menu function from the IO class.

For user choice 1, we call the input\_student\_data function from the IO class and get the user input.

For user choice 2, we call the output\_student\_course function from the IO class and present the user with the current registration data.

For user choice 3, we call the write\_data\_to\_file function from the FileProcessor class and save the data to the JSON file.

For user choice 4, we break out of the loop and close the program.

If the user choice isn’t 1, 2, 3, or 4 the program will display an “Invalid Option” message and ask the user to choose from one of the options from the menu of choices.

This is all shown in Figure 5.

A computer screen shot of text

Description automatically generated

***Figure 5. Main body of the script and calling on previously defined functions and classes.***

**Testing the code**

We run the code in both Command Prompt and PyCharm to ensure that it functions properly, and open the .csv file to make sure that the information is written properly. This is shown in Figure 6.

A screenshot of a computer

Description automatically generated ***Figure 6. Testing the code.***

**Summary**

After reading the provided materials and watching the instructional videos, I was able to perform my assignment as outlined in the steps above. The program created demonstrates my use of classes and functions as well as the concept of separation of concerns.