Name: Ezra Feyissa
Date: 05/21/2025
Course: Python 100

Assignment 05 – Create a Program Using Constants, Variables, while loop, error handling and Print Statement and save the user input and constants as a Json file.

Introduction

This assignment like assignment 01, 02, 03 and 04 uses constants, variables, user input () function and print statement. Assignment five requires using Json file to save student data. The code created also handles errors in a way that is easy for the user to understand them.

Tools and Technologies

The script was written on PyCharm and tested on Python IDLE and Terminal.

Program Structure

The script has a header containing a title, short description, change log, followed by a setup section and the body of the code.

Implementation Details

The goal of this assignment is to display a menu with several options, beginning with a prompt for the user to select an action. If the user chooses to register for a course, the program will collect and store the input data until it is recalled. if prompted, it will print the student's name along with the course name. If the user continues adding additional students and courses, the program will retain this new information alongside the previously entered data. The program also has the capability to save all inputs to a CSV file. To do so, the user simply selects the appropriate option from the menu to save the stored data.
If the Json file does not exist or the first name and last name the user enters starts with a number instead of letters, the code will let the user know what the issue is in a meaningful way.

The goal is accomplished by:

1. It starts by importing the JSON module to handle reading and writing data to a file.
2. Some variables and constants are set up, including the file name and a menu for the user.
3. The program tries to load any existing student data from a file. If the file doesn't exist, it gives a helpful error message.
4. Then it enters a loop where it keeps showing a menu until the user decides to exit.

5. If the user chooses option 1, they can register a new student by entering their name and course, and the data gets added to a list.
    - shows all students currently stored in the list.
    - saves the student data to a JSON file.
    - ends the program.
6. Throughout the program, errors like bad input or file problems are handled so the program doesn't crash.
7. It also makes sure files are properly closed after being used.

To add more context to the process explained above, a screenshot of the script along with the output is attached below.

```python
# ------------------------------------------------------------------------------------------ #
# Title: Assignment05
# Desc: This assignment demonstrates using dictionaries, files, and exception handling
#  Ezra Fevissa, 05/21/2025, Assignment05 script
# ------------------------------------------------------------------------------------------ #

#setup section

import json as js


# Define the Data Constants
MENU: str = '''
---- Course Registration Program ----
  Select from the following menu:
    1. Register a Student for a Course.
    2. Show current data.
    3. Save data to a file.
    4. Exit the program.
-----------------------------------------
'''
# Define the Data Constants
FILE_NAME: str = "Enrollments.json"

# Define the Data Variables and constants
student_first_name: str = ''  # Holds the first name of a student entered by the user.
student_last_name: str = ''  # Holds the last name of a student entered by the user.
course_name: str = ''  # Holds the name of a course entered by the user.
student_data: dict = {}  # one row of student data (TODO: Change this to a Dictionary)
students: list = []  # a table of student data
#csv_data: str = ''  # Holds combined CSV data. Note: Remove later since it is NOT needed with the JSON File
file = None  # Holds a reference to an opened file.
menu_choice: str  # Hold the choice made by the user.


# When the program starts, read the file data into a list of lists (table)
# Extract the data from the file
# file = open(FILE_NAME, "r")
# for row in file.readlines():
```

```
    try:

        file = open(FILE_NAME, 'r')
        students = js.load(file)
        file.close()
    except FileNotFoundError as e:
        print("Text file must exit before running this script!\n")
        print("Technical Error Message --")
        print(e, e.__doc__, type(e), sep='\n')
    except Exception as e:
        print("There was a non-specific error!\n")
        print("--Technical Error Message --")
        print(e, e.__doc__, type(e), sep = '\n')
    finally:
        if file.closed == False:
            file.close()



    # Present and Process the data
    while (True):

        # Present the menu of choices
        print(MENU)
        menu_choice = input("What would you like to do: ")

        # Input user data
        if menu_choice == "1":  # This will not work if it is an integer!
            try:
                student_first_name = input("Enter the student's first name: ")
                if not student_first_name.isalpha():
                    raise ValueError("The first name should not contain number.")
                student_last_name = input("What is the student's last name?")
            except student_last_name.isalpha():
                    raise ValueError("The last name should not contain number.")

            course_name = input("Please enter the name of the course: ")
            student_data = {"FirstName": student_first_name, "LastName": student_last_name, "CourseName":course_name}
```

```
    # Save the data to a file
    elif menu_choice == "3":
        file_obj = open(FILE_NAME, 'w')
        for student_data in student_list:
            file_obj.write(f'{student_data[0]},{student_data[1]},{student_data[2]}\n')
        file_obj.close()

    elif menu_choice == "4":
        break  # out of the loop

    else:
        print("Please only choose option 1, 2, 3, or 4")

print("Program Ended")
```

Assignment05.py

```
            students.append(student_data)
            print(f"You have registered {student_first_name} {student_last_name} for {course_name}.")
            continue

        # Present the current data
        elif menu_choice == "2":

            # Process the data to create and display a custom message
            print("-"*50)
            for student in students:
                message = '{} {} is enrolled in {}'
                print(message.format( *args: student ['FirstName'] , student['LastName'],student ['CourseName']))
            print("-"*50)
            continue

        # Save the data to a file
        elif menu_choice == "3":
            try:
                file = open(FILE_NAME, "w")
                js.dump(students, file)
                file.close()
                continue
            except TypeError as e:
                print("Please check that the data is a valid JSON format\n")
                print("-- Technical Error Message --")
                print(e, e.__doc__, type(e), sep = "\n")
            except Exception as e:
                print("-- Technical Error Message -- ")
                print("Built-In Python error info: ")
                print(e, e.__doc__, type(e), sep='\n')
            finally:
                if file.closed == False:
                    file.close()
            # for student in students:
            #     csv_data = f"{student[0]},{student[1]},{student[2]}\n"
            #     file.dump(students)
            # file.close()
                print("The following data was saved to file!")
                print("-" * 50)
```

```
                print("-" * 50)
                for student in students:
                    message = '{} {} is enrolled in {}'
                    print(message.format( *args: student ['FirstName'] , student['LastName'],student ['CourseName']))
                continue

        # Stop the loop
        elif menu_choice == "4":
            print("Program Ended")
            break  # out of the loop
    else:
        print("Please only choose option 1, 2, 3, or 4")

```

Running and Testing

This code was written in PyCharm and run and tested on Integrated Developed Environment (IDLE) and Terminal.

Summary

This assignment involved building a Python script that displays a menu-driven interface, allowing the user to register students for courses, display current student data, and optionally save the data to a JSON file. The program uses variables to store input such as student first name, last name, and course name. It makes use of loops to repeatedly show the menu, and conditional statements to process user selections. Student records are structured using dictionaries and stored in a list. The script also demonstrates file handling by reading existing data from a JSON file at the start and saving new data upon request. Exception handling is used to manage issues like missing files or invalid input. Overall, the program shows how to collect input, structure data, and manage persistent storage efficiently.