

Katie Debetaz

Nov 12, 2024

Foundations of Programming: Python

<https://github.com/kdebetaz/IntroToProg-Python-Mod05>

## Assignment 05: Advanced Collections and Error Handling

### Introduction

In this assignment, I will outline the steps I took to allow the user to choose options from a menu and then perform different actions based on the input given. I will first give details on how I loaded in the existing data from the json file and then added new data to the file. I will then share how I included structured error handling throughout the program.

### Interacting with JSON Files

The first thing I did to begin this program was read the existing data in a json file using the `.load()` function. I then assigned the data to its respective variables to be used throughout the program. Under the third menu choice, I updated the code to export the new student data to the json file. Although the process is similar to writing to a csv file, I used a different function, `dump()`, to write to this json file.

```
# Export data to json
elif menu_choice == "3":
    try:
        file = open(FILE_NAME, "w")
        json.dump(students, file)
        file.close()
        print('The following was saved to json:')
        for student in students:
            print(f"{student['FirstName']} {student['LastName']} is registered for {student['CourseName']}.")
    except Exception as e:
        print('Error exporting data to json')
    finally:
        if file and not file.closed:
            file.close()
```

### Error Handling

The other main difference between assignment 4 and 5 is the inclusion of structured error handling. I added try and except statements for reading and writing to the json file as well as when asking the user to input the student names. These will show the user the error if there is any issue with the json file interaction. Also, if the user inputs a name that is anything other than just letters, it will ask the user to ensure the entry is only alphabetic.

```
# Input for student name and course
if menu_choice == "1":
    try:
        student_first_name = input("Enter student's first name: ")
        if not student_first_name.isalpha():
            raise ValueError("First name must be alphabetic, please enter student's first name.")
        student_last_name = input("Enter student's last name: ")
        if not student_last_name.isalpha():
            raise ValueError("Last name must be alphabetic, please enter student's last name.")
        course_name = input("Enter the name of the course: ")
        student_data = {"FirstName":student_first_name,"LastName":student_last_name,"CourseName":course_name}
        students.append(student_data)
    except ValueError as e:
        print(e)
```

## Summary

This assignment demonstrates the steps taken to read existing data and then append new data to a json file. The program also includes error handling to make it more user friendly and able to display to the user some issues that may arise.