Noah Cooper
9/1/24
IT FDN 110 B Su 24: Foundations Of Programming: Python
Assignment 05

Assignment 05 - Advanced Collections and Error Handling

Intro

In this module, we learned about how to work with dictionaries, provide exception handling for unforeseen circumstances while running our script, and how to organize and upload our files to online repositories such as Github.

Learning Fundamentals

One of the major areas of focus for this module was with dictionaries. Dictionaries are similar to lists in that they can store multiple entries of data variables, but also feature keys. These are identifying values that are useful when converting dictionaries to spreadsheets. We learned how to add and remove data from dictionaries, and how to access data from files and convert it into a dictionary.

Another topic that was covered was JSON files. These are similar to CSV files, but are usually used for more complex data storage. JSON files use a hierarchical data structure, and are more easily readable for people. They are often used for configuration files, web API's, databases, and more. The module content gave us a glimpse into what it looks like when working with JSON files.

The topic of exception handling was also demonstrated with this module. When unexpected circumstances happen when a user is operating our script, we ideally want to be able to catch any errors rather than forcing the script to stop functioning. The goal of exception handling is to provide feedback as to what error occurred and to allow the script to continue running. We use the "try" statement to run our usual code, and if an error occurs with our operation, we use the "except" statement afterwards. This allows us to print an error statement of our own, and to print more technical details of the error by printing the exception as an object.

The last point of information in the module was about network file sharing. This has a variety of benefits, such as allowing for better collaboration, file backups, version control, documentation, and more. This module's assignment allowed us to practice using Github by uploading our assignment file to an online repository.

Assignment

The assignment expanded on the concepts of the last assignment, but with the key difference of utilizing dictionaries, as well as providing exception handling.

After opening the file for reading the data at the start of the script, we assign the values into a dictionary. This allows us to organize the read data with identifying keys, in this case “FirstName”, “LastName”, and “Class”. The data is then appended onto the students list, which we will use to properly re-save this data once the user selects option 3 to save to file.

The script now also makes use of exception handling techniques. If the file cannot be read for some reason, then the user will receive an error informing them. In addition, if the user tries to input invalid characters for the student’s name, such as numbers, they will also receive an error. The same goes for if the script cannot write to file when selecting option 3 to save the data. This is to ensure the script does not stop running due to an error, and give specific feedback as to what occurred.

Lastly, the assignment was uploaded to a Github repository. Practicing using Github has proven useful, as hosting files online is a good practice.

```
========= RESTART: C:\Users\Dr. Bones\Desktop\_Module05\assignment05.py ========
=== 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 program
Select an option: 1

Enter student's first name: Noah
Enter student's last name: Cooper1
Error: Unable to process data.
-- Technical Error Message --
Inappropriate argument value (of correct type).
Error: The last name should not contain numbers!
=== 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 program
Select an option:
```

Figure 1: The script running in the IDLE console. Note how the script provides an error for when the user inputs a number for the student’s last name.

Figure 2: The script running in PyCharm.

Conclusion

This module further expanded on the concepts of the last assignment, as well as providing insight into proper debugging and online file hosting techniques that are useful for all programmers.

While I had previously been struggling with the concepts of module 4, I feel more confident now going forward. I wanted to ensure that I fully understood the module lessons before continuing. I appreciate having the extra time to work through this week's assignment. I have already begun to study the module content for week 6. I hope to not have this put me behind schedule for the next assignment, and hope to upload each future assignment on time.