Assignment 07 – Classes and Objects

Student Name: Melissa Zheng

Date: June 04, 2025

Summary of Work

This assignment focused on using object-oriented programming in Python to manage student data. I implemented two custom data classes: 'Person' and 'Student'. The 'Student' class inherits from 'Person' and includes an additional attribute for the course name. I used properties to enforce input validation and implemented custom __str__ methods for clean output formatting.

Key Steps Completed

Reviewed the starter Python script and assignment requirements:
Carefully examined the provided Assignment07-Starter.py file and the accompanying documentation to fully understand the goals of the assignment, including the use of classes, object management, and file operations.

Created a Person class with validated name properties:
Implemented a Person class with first_name and last_name attributes. Used Python @property decorators to enforce validation, ensuring that only alphabetic strings are accepted for names.

Developed a Student class that inherits from Person:
Built a Student class that extends Person by adding a course_name property. This class encapsulates all the necessary data for student enrollment and demonstrates inheritance and method overriding.

Refactored the FileProcessor class for object serialization:
Updated the read_data_from_file and write_data_to_file methods to convert between Student objects and dictionary representations, allowing JSON-based persistence while maintaining object-oriented design.

Updated the IO class to interface with Student objects:
Rewrote input/output methods to interact with Student instances instead of raw dictionaries. This involved modifying display logic and input validation for better structure and reusability.

Added structured error handling:
Implemented try-except blocks to handle invalid user input and file I/O errors gracefully, providing meaningful feedback using the output_error_messages method for better user experience.

Thoroughly tested the program's functionality:
Verified that the application supports all required features—registering students, displaying data, saving to file, and loading from file—while ensuring clean user interaction and no runtime errors.

Published the final script and documentation to GitHub:
Uploaded both the completed Python script and this reflective document to a public GitHub repository, ensuring the work is accessible for peer review and version control.

Lessons Learned

Through this assignment, I gained a deeper understanding of class construction, inheritance, and data encapsulation using properties. I also practiced how to use exception handling in a user-friendly way and convert between object and dictionary representations for JSON file processing.

GitHub Repository

Melissa0407022/IntroToProg-Python-Mod07