Name: Cari Williams

Date: 29 November 2025

Course: Foundations Of Programming with Python

GitHub: https://github.com/carilynette/IntroToProg-Python-Mod08

Employee Ratings Application

Application Overview

The Employee Ratings application is a Python program that collects, displays, and saves employee performance review data. It reads and writes employee records to a JSON file, allows the user to enter new review information, and presents the data in a clean, easy-to-read format.

This final Module 8 assignment combines all major course concepts: modules, classes, properties, validation, functions, loops, file handling, and unit testing.

Program Structure

The application is divided into four main modules plus test files:

1) Data_classes.py – defines the program's structure. Contains default values
    - Person – the person class stores first and last name and contains validation that the names only contain alphabetic characters.
    - Employee – inherits from Person and adds review date and review rating.

b) Processing_classes.py – handles the reading and writing to JSON files. Includes error handling for file access, JSON formatting and permissions.
    - FIleProcessor.read_employee_data_from_file() reads JSON file, converts each row into an Employee object, validates the fields, and loads objects into a list.
    - File.Processor.write_employee_data_to_file() converts the object back into dictionaries and writes to the JSON file.

c) Presentation_classes.py – handles all user interaction
    - output_menu() – Displays the menu options.
    - input_menu_choice() – Gets and validates the user's menu choice.
    - input_employee_data() – Collects first name, last name, review date, and rating
    - output_employee_data() – Prints employee review information with rating descriptions (Leading, Strong, Solid, Building, Not Meeting Expectations).
    - output_error_messages() – Standardizes error formatting.

d) Main.py – the coordinator of the program.
   - Loads data at startup using FileProcessor.read_employee_data_from_file().
   - Displays the menu in a loop.
   - Processes menu options
   - Uses structured error handling around each menu option.

How the Program Works

StartUp

The program loads existing employee ratings from EmployeeRatings.json. If the file is missing, the program shows an error but continues to run with an empty list.

Menu Options

The user is presented with options to choose from:

- Show current employee ratings - Displays current employee ratings in an easy-to-read format. For example, John Doe was rated on 2025-11-30 and is rated as 4 (Strong)
- Enter new employee rating – prompts the user to enter first and last names, review date (YYYY-MM-DD) and review rating, 1-5. The data is stored as an Employee object.
- Save data to file – converts all data to dictionaries and writes to JSON file.
- Exit the program

Structured Error Handling is included in the program for:

- Invalid names (non-numeric)
- Invalid review date (wrong format or bad date)
- Invalid rating (not 1-5)
- File read/write issues
- Invalid menu choices

Unit Testing confirmed that the program behaves correctly and the validation works as expected. Three test modules are included: test_data_classes.py; test_processing_classes.py; test_presentation_classes.py.

Summary

This Employee Ratings application demonstrates a complete multi-module Python program featuring object-oriented design, JSON file handling, robust input validation, separation of concerns, structured error handling, and unit testing. It is functional, user-friendly, and follows the best practices developed throughout the course.