

Employee Ratings Application

Description

This project is a simple Employee Ratings Application built in Python. It allows users to manage employee data through a menu-driven interface. Users can view employee records, add new employees, and save data to a file. The system is divided into three main components:

- **Data Classes:** Handles data structures and constants.
- **Processing Classes:** Manages file I/O operations.
- **Presentation Classes:** Handles user input and output.

Features

- **View Employees:** Display a list of all employee records.
- **Add Employee:** Input new employee data and update the records.
- **Save Data:** Save the employee data to a file for persistence.
- **Exit Program:** Safely exit the system.

Table of Content

- [Employee management application](#)
- [Description](#)
- [Installation](#)
- [Usage](#)
- [Modules](#)
- [Contributing](#)
- [License](#)
- [Contact](#)

Installation

Follow these steps to get the development environment set up:

1. Clone the repository:

```
 git clone https://github.com/lorenasml/IntroToProg-Python-Mod08-FINAL.git
```

2. Navigate into the project folder:

► `cd IntroToProg-Python-Mod08-FINAL`

3. Ensure you have Python 3.13 installed.
4. Create or update the required data file (see configuration in the `data_classes` module).

Usage

1. Run the program:

► `python main.py`

2. The system will display a menu with the following options:

- **Option 1:** View Employee Data.
- **Option 2:** Add Employee Data.
- **Option 3:** Save Data to File.
- **Option 4:** Exit Program.

3. Follow the prompts to interact with the system:

- View all current employee records.
- Add new employees to the list.
- Save updates to the data file.

4. Exit the program by selecting Option 4.

Modules

The project is structured into the following modules:

data_classes

- Defines constants such as `FILE_NAME` and `MENU`.
- Structures employee data.

processing_classes

- Handles file operations:
- `read_employee_data_from_file()` : Reads employee data from the file.
- `write_employee_data_to_file()` : Writes employee data to the file.

presentation_classes

Manages user interface interactions:

- `output_menu()` : Displays the menu to the user.
- `input_menu_choice()` : Captures the user's menu selection.

- `output_employee_data()` : Displays the employee records.
- `input_employee_data()` : Captures new employee data from the user.
- `output_error_messages()` : Handles and displays errors.

Contributing

Contributions are welcome! To contribute:

1. Fork the repository.
2. Create a new branch:

```
git checkout -b feature-name
```

3. Make your changes and commit them:

```
git commit -m "Add feature-name"
```

4. Push your changes and create a pull request.

License

Contact

For questions or support, please contact:

Email: lorenasm@uw.edu

GitHub: [lorenasml](#)