

COMP1002 – Advanced Python Final Project Report

(Daily Mood Tracker)

Developer

NAME- SURNAME: Sena Uyar

ID: 232010020037

Introduction

This project is a desktop application developed individually as part of the COMP1002 – Advanced Python course.

The main goal is to provide a simple and practical system where users can easily select and save their daily mood.

Additionally, the application aims to help users build emotional awareness by allowing them to review their past mood entries.

Technologies Used

- **Python 3:** The entire project was developed using Python.
- **Tkinter:** Used to create the desktop graphical interface.
- **JSON:** Chosen for saving user data in a lightweight, readable format.
- **datetime:** Used for managing date and time information.
- **os:** Used for file checks and operations.

Implementation Details and Features

The application is designed following object-oriented programming (OOP) principles.

There are two main classes:

- **MoodEntry:** Represents a single mood entry. It stores the selected mood, emoji, and date.
- **MoodTracker:** The main class that creates the GUI, processes user input, and displays past entries.

The user interface is simple and designed to be user-friendly. On the main screen, the user is asked, “How are you feeling today?”. Several moods represented by emojis and labels are displayed as buttons. When the user selects a mood and clicks the “Save” button, their choice is saved to a file named `mood_data.json` along with the current date.

Additionally, the app lists the past 7 days' moods in chronological order. Only the date and mood name are shown, allowing the user to reflect on their recent emotional trend without needing charts or graphs.

Challenges & Solutions

- **GUI Design:** Designing a clean and intuitive interface with Tkinter took some time. Several iterations were made regarding color schemes, font sizes, and button layouts.
- **Saving and Reading Data:** Try-except blocks were added to ensure data is safely written to and read from the JSON file, even if the file doesn't exist initially.
- **Displaying Recent History:** To show only the last 7 days, `datetime` and `timedelta` modules were used to filter entries based on the date.

Conclusion & Future Work

The core features of the project were successfully implemented. Users can select their mood for the day and access a list of their recent entries.

Possible improvements for the future include:

- Suggesting songs based on the selected mood
- Letting users write a short note about how they feel
- Adding visual mood analysis charts
- Exporting data on a weekly or monthly basis

This project helped me apply my Python knowledge to a real-world problem and build something meaningful. It also showed me how powerful and effective a simple interface can be when it's built with the user in mind.