Abstract- Habit Tracker Project

Name: Jona Hajdari

Matriculation Number: 92123587

Module: Object-Oriented and Functional Programming with Python (DLBDSOOFPP01)

Project Summary

The Habit Tracker project was developed as the final portfolio task for the course *Object-Oriented and Functional Programming with Python*. The aim of the application was to enable users to build and maintain consistent habits by tracking daily or weekly goals through a command-line interface. Core functionality includes habit creation, task completion, streak calculation, and basic analytics.

Technical Implementation

To meet the module requirements, I implemented both Object-Oriented and Functional Programming techniques. The application's structure is built around classes such as Habit and HabitTracker, which handle the creation, completion, and saving of habits. Data is persisted between sessions using the JSON file format.

To fulfill the requirements for functional programming, I developed an external analytics.py module. This module analyzes habits using pure, stateless functions – such as get_longest_streak() and get_habits_by_periodicity() –making use of Python's filter(), lambda, and max() functions. These operate directly on the JSON data to calculate habit trends and streaks without relying on object state.

Challenges and Reflections

Although I have prior experience with Python, this project pushed me to apply my skills in a structured, multi-phase academic setting. The main challenge was ensuring a clean separation between object-oriented and functional programming logic while maintaining usability and consistency in the CLI.

Managing data-based logic for calculating daily and weekly streaks also required precise control using datetime. Additionally, designing pure, testable functions for analytics while keeping the program modular and readable was intellectually engaging, rather than technically difficult. This project strengthened my ability to architect systems with clean separation of concerns and to align real-world functionality with academic criteria.