# Habit-Tracker CLI (Object-Oriented & Functional Python)

Hi! I’m building this as my portfolio project for Object Oriented and Functional Programming with Python (DLBDSOOFPP01).

The brief: create a lightweight backend for a habit-tracking app—no GUI, just a clean command-line interface, solid OO design, functional analytics, JSON persistence, and unit tests.

---

* Features

Create / delete habits with daily or weekly periodicity

Complete\*\* a habit at any time; streak logic updates automatically

CLI analytics\*\* (all habits, filter by periodicity, longest streak overall / per habit)

JSON storage\*\* so data survives between runs

Fixture generator\*\* (`generate\_fixture\_data.py`) fills 5 sample habits over 4 weeks

Pytest suite\*\* for critical business rules

Pure-Python ≥ 3.7, zero external deps except `pytest` (dev only)

How to start? Mentioned step by step below!

bash

1) set up a virtual environment (recommended)

python -m venv venv

source venv/Scripts/activate # Windows

. venv/bin/activate # macOS / Linux

2) install requirements

pip install -r requirements.txt # only pytest is listed

3) (optional) generate demo data

python generate\_fixture\_data.py

4) run the CLI

python habittrackermain.py

**CLI Guide**

| **Main Menu** | **What it does** |
| --- | --- |
| 1 Create a new habit | Prompts for name + periodicity |
| 2 Complete a habit | Marks the habit done *now* |
| 3 View all habits | Prints each habit & current streak |
| 4 Analytics | Sub-menu for reports |
| 5 Save and Exit | Writes habits.json and quits |

* **Data Storage**

All state is stored in **habits.json** (project root).  
Structure: a list of habit objects with ISO-8601 timestamps.  
Swap in an SQLite layer later if needed—HabitManager isolates storage.

* **Running tests**

Initiate the test using “pytest”

The suite lives in tests/ and currently covers:

* streak calculation (daily / weekly)
* broken-habit detection
* analytics helpers

## Project structure

habit.py To Build Habit class (OO)

habit\_manager.py To Create / delete / persist habits

analytics.py For Functional utilities

habittrackermain.py For CLI entry point

generate\_fixture\_data.py

tests( below mentioned two files are the part of this folder.)

test\_habit.py

test\_analytics.py

requirements.txt

README.md