

Simple Python Budget Tracker

A robust, console-based application for managing personal finances with persistent data storage.

UJJWAL JALOTRA
25BAI10118

Project Overview

- ✓ **The Problem:** Tracking monthly expenses manually is tedious and prone to error.
- ✓ **The Solution:** A Python automation tool that digitizes the process.
- ✓ **Core Function:** Allows users to set budgets, log expenses, and view real-time balance updates.
- ✓ **Target Audience:** Individuals looking for a lightweight, CLI-based financial management tool.



Key Features



Data Persistence

Uses JSON to store budget and expense data locally. Your financial data survives even after the program closes.



Smart Tracking

Dynamically aggregates expenses by category. Adding "Food" twice sums the amounts automatically.



Budget Alerts

Provides instant feedback on your financial health, alerting you immediately if you exceed your monthly limit.

Technical Stack

- 🐍 **Python 3.x:** Core logic and flow control.
- 💾 **JSON Module:** Handles data serialization and file I/O operations.
- 🛡 **Error Handling:** Robust try-except blocks manage invalid inputs and file errors.
- **CLI Interface:** A clean, menu-driven loop for user interaction.

```
4  <head>
5    <meta charset="UTF-8" />
6    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7    <title>Style Demo</title>
8  </head>
9  <body>
10 <script>
11   function $initHighlight(block ,flags) {
12     //...some gibberish function
13     if (!!flags) {
14       try {
15         const someString = "123";
16         if (block.abc.className.search(/\bno-highlight\b/ != -1))
17           return processBlock(block.__proto__.function, true, 0xff);
18       } catch (e) {
19         for (let idx = 0 / 2; idx < classes.length; idx++) {
20           if (checkConditions(classes[i]) === undefined) return /\d+[\s]/g;
21         }
22       }
23     }
24   }
25 </script>
```

Application Data Flow



JSON Structure

The application relies on a structured JSON format to maintain state. This lightweight approach ensures portability and human-readability.

- 🔑 **Total Budget:** Stored as a float value.
- ☰ **Expenses:** A dictionary object mapping categories to aggregated amounts.
- ⟳ **Auto-Save:** The `save_data()` function triggers automatically after every critical update.



Future Enhancements

Visual Reporting

Implement Matplotlib or similar libraries to generate pie charts and bar graphs, visualizing spending habits directly within the application.

GUI Development

Transition from the Command Line Interface (CLI) to a Graphical User Interface (GUI) using Tkinter or PyQt for a more modern user experience.

Questions?

Thank you for reviewing the Python Budget Tracker project.

Image Sources



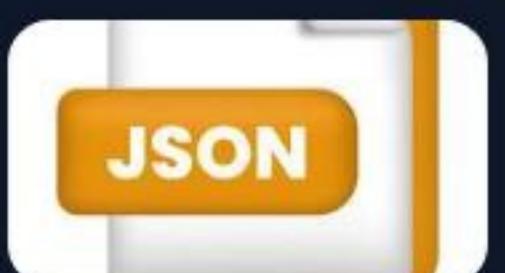
[https://elements-resized.envatousercontent.com/elements-video-cover-images/a1c4ee92-0d4a-47ea-913b-791ab69ba57d/video_preview/video_preview_0000.jpg?w=500&cf_fit=co
ver&q=85&format=auto&s=efaf5d5e060e9dda7f5fb4f3e7ee468ad7d9c7b5e40c64f1f7875567419deced](https://elements-resized.envatousercontent.com/elements-video-cover-images/a1c4ee92-0d4a-47ea-913b-791ab69ba57d/video_preview/video_preview_0000.jpg?w=500&cf_fit=cover&q=85&format=auto&s=efaf5d5e060e9dda7f5fb4f3e7ee468ad7d9c7b5e40c64f1f7875567419deced)

Source: elements.envato.com



https://miro.medium.com/max/661/1*QtxAF_v7uu19JhR4Zm-n-A.png

Source: plainenglish.io



https://img.freepik.com/premium-vector/json-file-format-icon-json-file-format-3d-render-icon-json-file-format-document-color-icon-vector_300191-1312.jpg?semt=ais_incoming&w=740&q=80

Source: www.freepik.com



https://img.freepik.com/premium-photo/abstract-glowing-financial-chart-dark-background-stock-market-economy-trends_715671-4148.jpg

Source: www.freepik.com