**Personal Expense Tracker – Project Documentation**

**Description**

The Personal Expense Tracker is a command-line Python application that allows users to log, view, and manage their daily expenses, track their spending habits, and ensure they stay within a predefined monthly budget. The tool also enables saving and loading expense data via a CSV file, allowing the user to resume from where they left off.

**Objectives**

Design and implement a personal expense tracking tool.

Allow users to log expenses with category, amount, description, and date.

Implement file-based persistence using CSV files to store and retrieve expenses.

Create an interactive menu to navigate various features easily.

**Features Implemented**

**Add Expense**

Prompts the user to enter a date, category, amount, and description for the expense. Input validation ensures proper formatting (e.g., date and numeric checks).

**View Expenses**

Displays all the expenses stored in memory in a user-friendly format with enumerated entries.

Set Budget

Asks the user to enter a monthly budget and compares it against total expenses to indicate whether the user has overspent, underspent, or matched the budget.

**Save to CSV File**

Saves all current expenses into a CSV file (Project.csv) with headers for easy viewing and analysis.

**Load from CSV File**

Automatically loads saved expenses from the CSV file when the program starts, so the user can continue seamlessly.

**File Structure**

**budget.py** – Main Python script containing all logic and functions.

**Project.csv** – The data file where all expenses are stored.

**Sample Expense Format (CSV)**

date,category,amount,description

2024-07-01,food,10.5,Lunch at cafeteria

2024-07-02,transport,3.0,Bus fare

**How to Run**

Ensure Python 3 is installed.

**Run the program:**

python budget.py

**Limitations & Future Enhancements**

Currently uses CSV for persistence; future upgrades can use SQLite or JSON.

No GUI; could be enhanced with Tkinter or a web-based frontend.

Categories could be made selectable from a predefined list to avoid typos.

**Conclusion**

This project demonstrates how basic file handling, data validation, and user interaction in Python can be combined

to solve real-world problems. It builds a foundation for future enhancements in personal finance and data tracking

applications.

Sample Screenshots









