📘 PERSONAL FINANCE MANAGER

Project Documentation

1. PROJECT OVERVIEW & OBJECTIVES

🔹 Project Title

Personal Finance Manager (Advanced Version)

🔹 Project Overview

The Personal Finance Manager is a **menu-driven Python application** designed to help users **track daily expenses, analyze spending patterns, and manage budgets efficiently**.
It uses **object-oriented programming, file handling, and data analysis techniques** to store and process financial data.

🔹 Objectives

- To help users record and manage personal expenses
- To analyze expenses using reports and charts
- To practice Python OOP, CSV file handling, and modular coding
- To build a real-world, beginner-to-intermediate level Python project
- To implement backup, restore, and export functionalities

2. SETUP & INSTALLATION GUIDE

🔹 System Requirements

- Operating System: Windows
- Python Version: **Python 3.9 or higher**
- IDE: **Visual Studio Code**

🔹 Step-by-Step Installation

Step 1: Install Python

Download from:
👉 https://www.python.org/downloads/

Verify installation:

python --version

Step 2: Install VS Code

Download from:
👉 https://code.visualstudio.com/

Install extensions:

- Python
- Pylance

Step 3: Install Required Libraries

pip install matplotlib

Step 4: Project Folder Structure

personal_finance_manager/

|

├── main.py

├── expense.py

├── utils.py

├── file_manager.py

├── reports.py

├── menu.py

├── charts.py

|

├── data/

| └── expenses.csv

|

├── backups/

| └── expenses_backup.csv

|

├── exports/

| └── summary_report.csv

Step 5: Run the Project

python main.py

3. USER MANUAL (HOW TO USE THE APPLICATION)

🔹 Main Menu Options

1. Add Expense
2. View Expenses
3. Generate Report
4. Backup Data
5. Restore Data
6. View Charts
7. Exit

🔹 Add Expense

- Enter amount, category, date, and description
- Input validation ensures correct data
- Expense is saved permanently in CSV file

🔹 View Expenses

- Displays all saved expenses in readable format

🔹 Generate Report

- Shows:
  - Total expenses
  - Average expenses
  - Category-wise expenses
- Automatically exports summary to CSV file

🔹 View Charts

- Category-wise bar chart
- Monthly expense trend line chart

🔹 Backup & Restore

- Backup saves data to backups/expenses_backup.csv
- Restore recovers data in case of accidental deletion

4. CODE STRUCTURE EXPLANATION

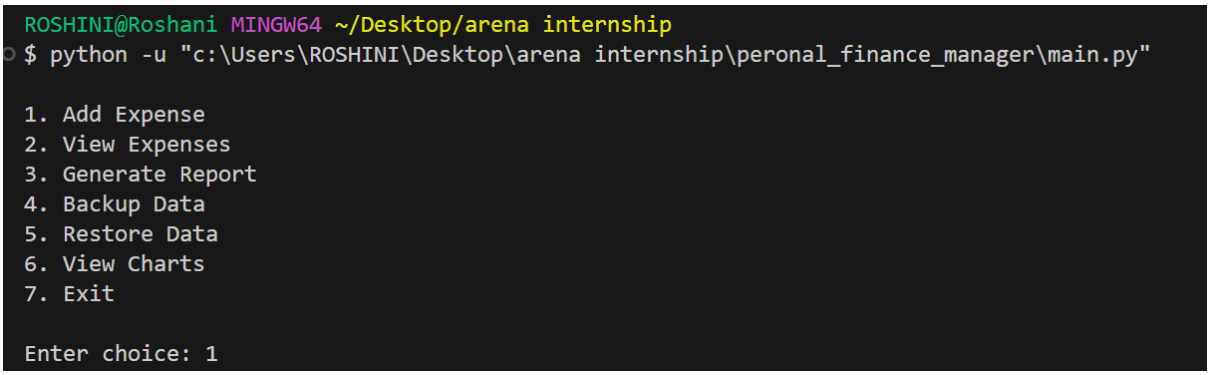
File Name	Description
main.py	Entry point, controls program flow
expense.py	Expense class (OOP implementation)
utils.py	Input validation & budget checking
file_manager.py	CSV operations, backup, restore, export
reports.py	Expense calculations and summaries
menu.py	Command-line menu interface
charts.py	Data visualization using matplotlib

---

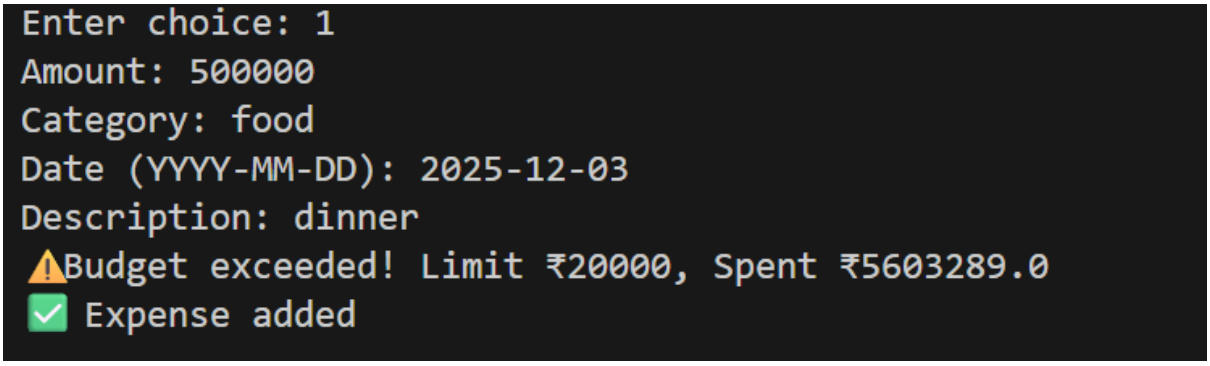
5. SCREENSHOTS OF WORKING APPLICATION

📸 **Include the following screenshots in screenshots/ folder:**

1. Application start menu



2. Adding an expense



3. Viewing expenses

```
1. Add Expense
2. View Expenses
3. Generate Report
4. Backup Data
5. Restore Data
6. View Charts
7. Exit

Enter choice: 2
500.0 food 2025-10-11 dinner
4000.0 food 2025-12-23 dinner
5098789.0 food 2013-12-04 dinner
500000.0 food 2025-12-03 dinner
```

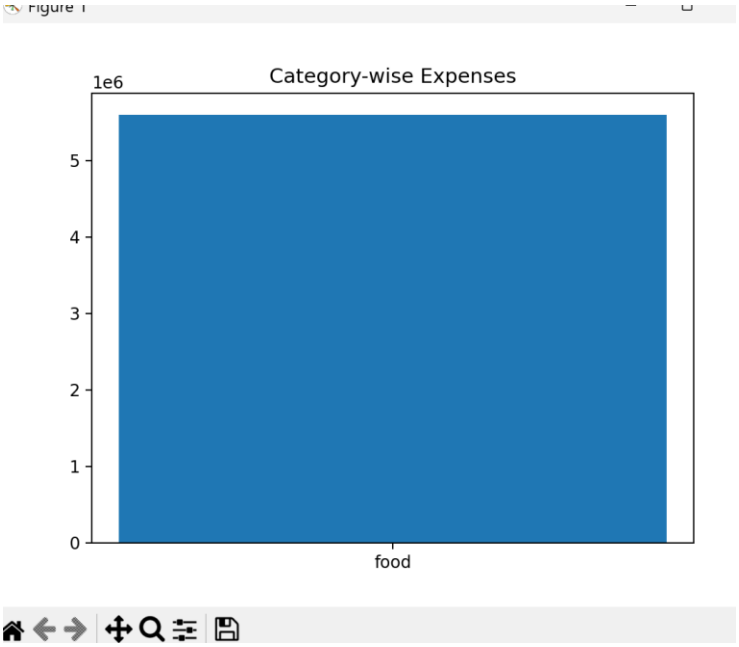
4. Report generation

```
1. Add Expense
2. View Expenses
3. Generate Report
4. Backup Data
5. Restore Data
6. View Charts
7. Exit

Enter choice: 3

📊 REPORT
Total: ₹5603289.0
Average: ₹1400822.25
food: ₹5603289.0
✅ Summary exported
```

5. Category-wise bar chart



6. TECHNICAL REQUIREMENTS – HOW THEY ARE MET

Requirement	Implementation
Expense class	expense.py using OOP
CSV persistence	file_manager.py
Error handling	try-except & validation
Menu system	menu.py
Data validation	utils.py
Modular structure	Separate files
Reports	reports.py
Backup & restore	file_manager.py
Charts	charts.py (matplotlib)
Export	CSV summary export

7. TESTING OVERVIEW

- **Types of Testing Performed**
  - Manual testing
  - Input validation testing
  - Functional testing
  - File handling testing

8. LIMITATIONS & FUTURE ENHANCEMENTS

- **Limitations**
  - CLI-based (no GUI)
  - Single-user system
- **Future Enhancements**
  - GUI using Tkinter
  - Cloud database integration

- Multi-user login system
- Mobile application version