

Personal Finance Manager

1. Introduction

The **Personal Finance Manager** is a Python-based command-line application that helps users manage their daily income and expenses.

It provides features such as adding expenses, adding income, viewing reports, generating summaries, and backing up data.

This user manual explains how to use every feature in the app.

2. Starting the Application

How to Run

1. Open Command Prompt / Terminal.
2. Navigate to your project folder:

```
cd personal-finance-manager
```

3. Run the application:

```
python main.py
```

You will see the main menu displayed.

3. Main Menu Options

When the program starts, the following menu appears:

```
MAIN MENU
1. Add Expense
2. View All Expenses
3. Add Income
4. View All Income
5. View Category-wise Expense Summary
6. Monthly Income Report
7. Income vs Expense Overview
8. Backup Data
9. Exit
```

4. Feature Guide (Step-by-Step)

4.1 Add New Expense

Use this option to record any spending.

You will be asked to enter:

- **Date (YYYY-MM-DD)**
- **Category** (Food, Transport, Shopping, etc.)
- **Amount**
- **Note** (optional)

Example Input:

```
Date: 2025-11-10
Category: shopping
Amount: 1000
Note: Festival
```

✓ Expense added successfully!

```
PS C:\Users\safe\Desktop\personal-finance-manager> python main.py
===== PERSONAL FINANCE MANAGER =====
1. Add New Expense
2. View All Expenses
3. Add New Income
4. View All Income
5. View Category-wise Expense Summary
6. Monthly Income Report
7. Income vs Expense Overview (Net Savings)
8. Backup Data
9. Exit
Enter your choice (1-9): 1
Enter date (YYYY-MM-DD): 2025-11-10
Enter category: shopping
Enter amount: 1000
Note (optional): festival
✓ Expense added successfully!
```

4.2 View All Expenses

Displays all saved expenses in a table format:

Example:

0	2025-01-03	Food	150	Breakfast at cafe
1	2025-01-04	Transport	40	Bus ticket
2	2025-01-05	Groceries	760	Weekly vegetables
3	2025-01-06	Shopping	1200	Clothing items

☞ Insert screenshot:

Enter your choice (1-9): 2					
All Expenses:					
	Date	Category	Amount	Note	
0	2025-01-03	Food	150	Breakfast at cafe	
1	2025-01-04	Transport	40	Bus ticket	
2	2025-01-05	Groceries	760	Weekly vegetables	
3	2025-01-06	Shopping	1200	Clothing items	
4	2025-01-08	Entertainment	300	Movie with friends	
5	2025-01-09	Food	220	Lunch at restaurant	
6	2025-01-12	Health	450	Medicines	
7	2025-01-14	Bills	980	Electricity bill	
8	2025-01-15	Transport	90	Auto fare	
9	2025-01-16	Groceries	1450	Monthly essentials	
10	2025-01-18	Shopping	650	Beauty products	
11	2025-01-20	Travel	600	Train ticket	
12	2025-01-22	Food	180	Snacks and juice	
13	2025-01-25	Entertainment	1100	Concert ticket	
14	2025-01-28	Bills	300	Water bill	
15	2025-02-01	Food	200	Dinner at cafe	
16	2025-02-03	Transport	500	Fuel	
17	2025-02-05	Shopping	1800	Shoes purchase	
18	2025-02-08	Entertainment	850	Theme park entry	

4.3 Add New Income

Use this option to record income such as salary, freelance payments, etc.

You will be asked to enter:

- Date (YYYY-MM-DD)
- Source
- Amount
- Note (optional)

Example:

Date: 2025-12-08
Source: Freelance Work
Amount: 5000
Note: Website project

✓ Income added successfully!

☞ Insert screenshot:

```
===== PERSONAL FINANCE MANAGER =====
1. Add New Expense
2. View All Expenses
3. Add New Income
4. View All Income
5. View Category-wise Expense Summary
6. Monthly Income Report
7. Income vs Expense Overview (Net Savings)
8. Backup Data
9. Exit
Enter your choice (1-9): 3
Enter date (YYYY-MM-DD): 2025-12-01
Enter income source: freelance
Enter amount: 100
Note (optional): project
✓ Income added successfully!
```

4.4 View All Income

Displays all income entries saved in the system.

Example:

```
0 2025-01-01      Salary  35000  Monthly salary credited
1 2025-01-05      Freelance 5000  Website design project
2 2025-01-12      Gifts    1500  Birthday gift from friend
3 2025-01-18  Online Selling 2200  Sold old books
```

 [Insert screenshot: view_income.png](#)

```
Enter your choice (1-9): 4



| All Income: |            |                |        |                           |
|-------------|------------|----------------|--------|---------------------------|
|             | Date       | Source         | Amount | Note                      |
| 0           | 2025-01-01 | Salary         | 35000  | Monthly salary credited   |
| 1           | 2025-01-05 | Freelance      | 5000   | Website design project    |
| 2           | 2025-01-12 | Gifts          | 1500   | Birthday gift from friend |
| 3           | 2025-01-18 | Online Selling | 2200   | Sold old books            |
| 4           | 2025-02-01 | Salary         | 35000  | Monthly salary credited   |
| 5           | 2025-02-06 | Interest       | 800    | Savings interest credited |
| 6           | 2025-02-12 | Freelance      | 6000   | Social media work         |
| 7           | 2025-02-20 | Bonus          | 2500   | Performance bonus         |
| 8           | 2025-12-01 | freelance      | 100.0  | project                   |


```

4.5 Category-wise Expense Summary

This feature shows the **total amount spent** in each category.

Example:

```
Bills           1280.0
Entertainment   2250.0
Food            750.0
Groceries       2210.0
Health           750.0
```

A bar chart will also be displayed (if matplotlib is installed).

☞ Insert screenshot: category_summary.png

```
Enter your choice (1-9): 5

[H] Expense Category-wise Summary:
category
Bills           1280.0
Entertainment   2250.0
Food            750.0
Groceries       2210.0
Health           750.0
Shopping         3650.0
Transport        630.0
Travel           600.0
shopping         1000.0
Name: amount, dtype: float64
```

4.6 Monthly Income Report

Shows all income and expense totals grouped by month.

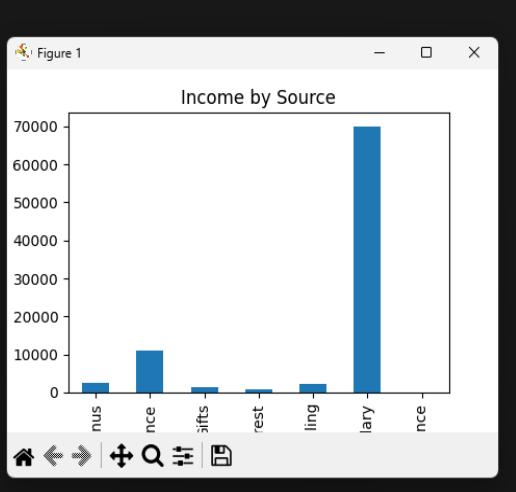
Example:

```
December 2025:
Bonus           2500.0
Freelance       11000.0
Gifts            1500.0
Interest          800.0
```

☞ Insert screenshot: monthly_report.png

```
==== PERSONAL FINANCE MANAGER ====
1. Add New Expense
2. View All Expenses
3. Add New Income
4. View All Income
5. View Category-wise Expense Summary
6. Monthly Income Report
7. Income vs Expense Overview (Net Savings)
8. Backup Data
9. Exit
Enter your choice (1-9): 6

Income Summary by Source:
source
Bonus      2500.0
Freelance   11000.0
Gifts       1500.0
Interest    800.0
Online Selling 2200.0
Salary      70000.0
freelance   100.0
Name: amount, dtype: float64
```



4.7 Income vs Expense Overview (Net Savings)

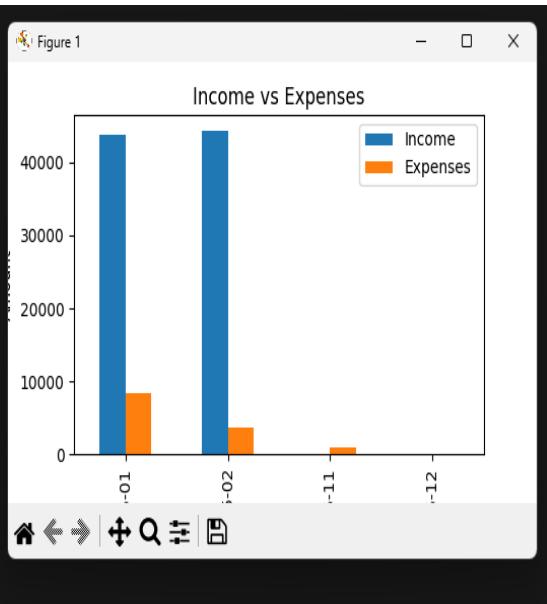
Shows your overall financial health:

- Total Income
- Total Expenses
- Net Savings
- Bar graph comparison (optional)

☞ Insert screenshot: income_vs_expense.png

```
==== PERSONAL FINANCE MANAGER ====
1. Add New Expense
2. View All Expenses
3. Add New Income
4. View All Income
5. View Category-wise Expense Summary
6. Monthly Income Report
7. Income vs Expense Overview (Net Savings)
8. Backup Data
9. Exit
Enter your choice (1-9): 7

Monthly Balance:
  Income Expenses
month
2025-01 43700.0 8470.0
2025-02 44300.0 3650.0
2025-11 0.0 1000.0
2025-12 100.0 0.0
```



4.8 Backup Data

Creates a **backup copy** of your CSV files inside the **backup/** folder.

Backup files:

- expenses_backup.csv
- income_backup.csv

You will see:

Backup created successfully!

4.9 Exit Program

Select this option to safely exit the application.

Thank you for using Personal Finance Manager!

5. File Storage Details

- **data/expenses.csv** — stores all expense entries
 - **data/income.csv** — stores all income entries
 - **backup/** — stores backup copies
 - Files are automatically updated after every entry
-

6. Tips for Best Use

- ✓ Enter dates in **YYYY-MM-DD** format
 - ✓ Backup data regularly
 - ✓ Use clear notes (e.g., "Lunch", "Petrol", "Freelance project")
 - ✓ View reports weekly to track spending habits
-

7. Troubleshooting

Issue	Cause	Solution
App not starting	Missing libraries	Run <code>pip install -r requirements.txt</code>
Wrong date format	Entered 12/03/25	Use YYYY-MM-DD
No graph appearing	Missing matplotlib	Install using pip
CSV file not showing data	File deleted/empty	Add new expense/income

8. Conclusion

Your Personal Finance Manager is ready to help you take control of your money. It is simple, fast, and effective for daily tracking and financial analysis.