

Expense Tracker Project

Project Requirements and Features

Project Objectives:

- Develop a system that allows users to input their daily expenses.
- Implement a mechanism to store and manage the entered expense data.
- Categorize expenses into different categories for better organization.
- Provide users with insights into their spending patterns, such as monthly summaries and category-wise expenditure.
- Create a user-friendly interface for a seamless user experience.
- Implement error handling to ensure the application can handle unexpected inputs gracefully.
- Document the code effectively to demonstrate clarity and understanding.

Requirements and Features:

- User Input: Allow users to input their daily expenses, including the amount spent and a brief description.
- Data Storage: Use appropriate data structures or file handling techniques to store and retrieve expense data.
- Expense Categories: Implement the ability for users to categorize their expenses (e.g., food, transportation, entertainment).
- Data Analysis: Provide users with the option to view summaries of their monthly expenses and category-wise expenditure.
- User Interface: Create a simple and intuitive user interface to interact with the Expense Tracker.
- Error Handling: Include error handling mechanisms to address potential issues during user interaction.
- Documentation: Provide clear documentation for your code, explaining the logic behind key functions and overall program structure.

Python Code

```
import json
import os
from datetime import datetime

data_file = "expenses.json"

def load_expenses():
    if os.path.exists(data_file):
        with open(data_file, "r") as file:
            return json.load(file)
    return []

def save_expenses(expenses):
    with open(data_file, "w") as file:
        json.dump(expenses, file, indent=4)

def add_expense():
    amount = float(input("Enter expense amount: "))
    category = input("Enter expense category (Food, Transport, Entertainment, etc.): ")
    description = input("Enter expense description: ")
    date = datetime.now().strftime("%Y-%m-%d")

    expenses = load_expenses()
    expenses.append({"date": date, "amount": amount, "category": category, "description": description})
    save_expenses(expenses)
    print("Expense added successfully!\n")

def view_expenses():
    expenses = load_expenses()
    if not expenses:
        print("No expenses recorded yet.\n")
```

```

        return

    for exp in expenses:
        print(f"{exp['date']} | {exp['category']} | Rs. {exp['amount']} | {exp['description']}")
    print()

def analyze_expenses():
    expenses = load_expenses()
    if not expenses:
        print("No expenses recorded to analyze.\n")
        return

    monthly_total = {}
    category_total = {}

    for exp in expenses:
        month = exp['date'][:7] # Extract YYYY-MM format
        monthly_total[month] = monthly_total.get(month, 0) + exp['amount']
        category_total[exp['category']] = category_total.get(exp['category'], 0) + exp['amount']

    print("Monthly Expense Summary:")
    for month, total in monthly_total.items():
        print(f"{month}: Rs. {total}")

    print("\nCategory-wise Expense Summary:")
    for category, total in category_total.items():
        print(f"{category}: Rs. {total}")
    print()

def main():
    while True:
        print("\nExpense Tracker")
        print("1. Add Expense")
        print("2. View Expenses")
        print("3. Analyze Expenses")
        print("4. Exit")

        choice = input("Choose an option: ")
        if choice == "1":
            add_expense()
        elif choice == "2":
            view_expenses()
        elif choice == "3":
            analyze_expenses()
        elif choice == "4":
            print("Exiting... Goodbye!")
            break
        else:
            print("Invalid choice, please try again.\n")

if __name__ == "__main__":
    main()

```

Sample Output

```

Expense Tracker
1. Add Expense
2. View Expenses
3. Analyze Expenses
4. Exit
Choose an option: 1

Enter expense amount: 500

```

Enter expense category (Food, Transport, Entertainment, etc.): Food
Enter expense description: Lunch
Expense added successfully!

Choose an option: 2

2025-03-04 | Food | Rs. 500 | Lunch

Choose an option: 3

Monthly Expense Summary:
2025-03: Rs. 500

Category-wise Expense Summary:
Food: Rs. 500

Choose an option: 4
Exiting... Goodbye!