

Expense Tracker

The Expense Tracker project is designed to reinforce your understanding of Python programming concepts and enhance your skills in building practical applications. In this project, you will be developing an Expense Tracker application that allows users to manage their expenses efficiently.

This real-world application will involve handling data, user input, and implementing key functionalities.

Code:

```
import json
from datetime import datetime

class ExpenseTracker:
    def __init__(self): # Corrected the method name to __init__
        self.expenses = []
        self.categories = set()

    def add_expense(self, amount, description, category):
        expense = {
            'date': datetime.now().strftime('%Y-%m-%d %H:%M:%S'),
            'amount': amount,
            'description': description,
            'category': category
        }
        self.expenses.append(expense)
        self.categories.add(category)
        print("Expense added successfully!")

    def view_expenses(self):
        if not self.expenses:
            print("No expenses recorded yet.")
        else:
            for expense in self.expenses:
                print(f>Date: {expense['date']}, Amount: ${expense['amount']},
Description: {expense['description']}, Category: {expense['category']}")

    def monthly_summary(self):
        monthly_expenses = {}
        for expense in self.expenses:
            month = expense['date'][:7]
```

```

        if month in monthly_expenses:
            monthly_expenses[month] += expense['amount']
        else:
            monthly_expenses[month] = expense['amount']
    print("Monthly Summary:")
    for month, total in monthly_expenses.items():
        print(f"{month}: ${total}")

def category_summary(self):
    category_expenses = {}
    for expense in self.expenses:
        category = expense['category']
        if category in category_expenses:
            category_expenses[category] += expense['amount']
        else:
            category_expenses[category] = expense['amount']
    print("Category-wise Summary:")
    for category, total in category_expenses.items():
        print(f"{category}: ${total}")

def save_to_file(self, filename):
    with open(filename, 'w') as file:
        json.dump(self.expenses, file)
    print(f"Expenses saved to {filename}")

def load_from_file(self, filename):
    try:
        with open(filename, 'r') as file:
            self.expenses = json.load(file)
            self.categories = set(expense['category'] for expense in
self.expenses)
        print(f"Expenses loaded from {filename}")
    except FileNotFoundError:
        print("File not found. Starting with an empty expense list.")

def main():
    tracker = ExpenseTracker()
    tracker.load_from_file('expenses.json')

    while True:
        print("\nExpense Tracker Menu:")
        print("1. Add Expense")
        print("2. View Expenses")
        print("3. View Monthly Summary")
        print("4. View Category-wise Summary")
        print("5. Save and Exit")
        choice = input("Enter your choice: ")

```

```
    if choice == '1':
        amount = float(input("Enter amount spent: "))
        description = input("Enter a brief description: ")
        category = input("Enter category: ")
        tracker.add_expense(amount, description, category)
    elif choice == '2':
        tracker.view_expenses()
    elif choice == '3':
        tracker.monthly_summary()
    elif choice == '4':
        tracker.category_summary()
    elif choice == '5':
        tracker.save_to_file('expenses.json')
        print("Exiting the Expense Tracker. Goodbye!")
        break
    else:
        print("Invalid choice. Please try again.")

if __name__ == "__main__": # Corrected the if condition here
    main()
```