Expense Tracker Project - Python Capstone

Objective: Build a command-line application that records daily expenses and provides summaries by category or date.

Skills Reinforced:

- File handling (reading/writing CSV)
- Data structures (lists, dictionaries)
- Basic data analysis and formatting

Step 1: Project Setup

Create a new Python file named **expense_tracker.py**. Import necessary modules such as csv and datetime.

Step 2: Define File Structure

Create a CSV file named **expenses.csv** if it doesn't exist. Each record should include: Date, Category, Description, and Amount.

Step 3: Add Expense Function

Write a function add_expense() that appends new expense entries to the CSV file.

Step 4: View Summary

Implement a function view_summary() that reads from the CSV and displays totals grouped by category or date.

Step 5: User Menu

Add a loop to present a simple text-based menu allowing users to add expenses, view summaries, or exit.

Sample Code Snippet:

```
import csv from datetime import datetime def add_expense(): date =
datetime.now().strftime('%Y-%m-%d') category = input('Enter category: ') description =
input('Enter description: ') amount = float(input('Enter amount: ')) with
open('expenses.csv', 'a', newline='') as file: writer = csv.writer(file)
writer.writerow([date, category, description, amount]) print('Expense added
successfully!') def view_summary(): with open('expenses.csv', 'r') as file: reader =
csv.reader(file) expenses = list(reader) total = 0 for row in expenses: print(row) total
+= float(row[3]) print(f'Total Expenses: {total}') while True: print('\nExpense Tracker
Menu') print('1. Add Expense') print('2. View Summary') print('3. Exit') choice =
input('Choose an option: ') if choice == '1': add_expense() elif choice == '2':
```

```
view_summary() elif choice == '3': break else: print('Invalid choice, please try
again.')
```

Next Steps:

Enhance the project by adding data visualization using matplotlib or exporting summaries to Excel using pandas.