Expense Tracker Program

Problem Statement

Create a simple **Expense Tracker** program that allows users to:

1. Add expenses (date, category, amount).
2. View all expenses.
3. View expenses by category.
4. Exit the program.

Requirements:

Menu-Driven Interface:

Display a menu with the following options:

Add a new expense.

View all expenses.

View expenses by category.

Exit.

Data Storage:

Use a list to store expense records.

Each expense record should be a dictionary with the following keys:

date: Date of the expense (string).

Category: Category of the expense (e.g., Food, Transportation).

amount: Amount spent (float).

Functionality:

Add a new expense: Allow the user to input the date, category, and amount for a new expense.

View all expenses: Display all expenses in a simple format.

View expenses by category: Allow the user to view expenses for a specific category.

Error Handling:

Handle invalid inputs (e.g., non-numeric amounts) gracefully.

Sample Output

```
Expense Tracker Menu:
1. Add a new expense
2. View all expenses
3. View expenses by category
4. Exit
Enter your choice: 1

Enter date (e.g., 2023-10-15): 2023-10-15
Enter category (e.g., Food, Transportation): Food
Enter amount: 25.50
Expense added successfully!

Expense Tracker Menu:
1. Add a new expense
2. View all expenses
```

```
3. View expenses by category
4. Exit
Enter your choice: 2

All Expenses:
Date        Category      Amount
---------------------------------
2023-10-15  Food          25.50
```

Learning Outcomes

- Practice using **lists** and **dictionaries** for data storage.
- Learn how to create a **menu-driven program**.
- Gain experience in **input validation** and **error handling**.
- Build a simple, functional program that can be expanded later

Enhance the Program: (optional)

- Add a feature to calculate the total expenses.

- Allow the user to delete an expense.

- Save and load expenses to/from a file.