Project-3

Project Name: Expense Tracker Date:03-07-24

Name: Hasanapuram Anilkumar

**Project Overview**

The Expense Tracker project aims to create a robust and user-friendly application that allows users to manage and monitor their expenses efficiently. This program will serve as a hands-on experience in applying Java programming concepts to develop a functional and practical software solution

**Project Objectives**

To accurately outline the scope of work required for a project, it is crucial to first identify its objectives. Pinpointing what the project hopes to accomplish will assist in determining its inclusions and limitations.

* **Project Overview Understanding Java Fundamentals:** Strengthen your grasp on core Java concepts such as classes, objects, inheritance, and exception handling.
* **Data Handling:** Implement data structures and algorithms to manage and organize expense data effectively.
* **User Interface Design:** Develop a simple and intuitive user interface for users to input, edit, and view their expenses.
* **File Handling:** Utilize file I/O operations to save and load expense data, ensuring persistence across sessions.
* **Exception Handling:** Implement robust error handling to enhance the reliability of the application.

Requirements and Features

* **User Registration:** Allow users to create accounts to personalize and secure their expense data.
* **Expense Entry:** Enable users to input details of their expenses, including the date, category, and amount.
* **Expense Listing:** Provide a clear and organized view of all entered expenses, with the ability to sort or filter.
* **Category-wise Summation:** Implement functionality to calculate and display the total expenses for each category.
* **Persistence:** Save and load expense data to and from files for a seamless user experience across sessions.

**INPUT CODE FOR EXPENSE TRACKER**

**package** oops;

**import** java.io.\*;

**import** java.util.\*;

// Expense class to represent each individual expense

**class** Expense

{

**private** String date;

**private** String category;

**private** **double** amount;

**public** Expense(String date, String category, **double** amount)

{

**this**.date = date;

**this**.category = category;

**this**.amount = amount;

}

**public** String getDate()

{

**return** date;

}

**public** String getCategory()

{

**return** category;

}

**public** **double** getAmount()

{

**return** amount;

}

@Override

**public** String toString()

{

**return** "Date: " + date + ", Category: " + category + ", Amount: $" + amount;

}

}

// ExpenseTracker class to manage expenses and user interactions

**public** **class** User

{

**private** List<Expense> expenses;

**private** Scanner scanner;

**public** User()

{

expenses = **new** ArrayList<>();

scanner = **new** Scanner(System.***in***);

}

// Method to add a new expense

**public** **void** addExpense()

{

System.***out***.println("Enter expense details:");

System.***out***.print("Date (MM-DD-YYYY): ");

String date = scanner.next();

System.***out***.print("Category: ");

String category = scanner.next();

System.***out***.print("Amount: $");

**double** amount = scanner.nextDouble();

Expense expense = **new** Expense(date, category, amount);

expenses.add(expense);

System.***out***.println("Expense added successfully!");

}

// Method to display all expenses

**public** **void** displayExpenses()

{

**if** (expenses.isEmpty())

{

System.***out***.println("No expenses entered yet.");

}

**else**

{

System.***out***.println("List of Expenses:");

**for** (Expense expense : expenses) {

System.***out***.println(expense);

}

}

}

// Method to calculate total expenses for a given category

**public** **void** calculateCategoryTotal(String category)

{

**double** total = 0;

**for** (Expense expense : expenses)

{

**if** (expense.getCategory().equalsIgnoreCase(category))

{

total += expense.getAmount();

}

}

System.***out***.println("Total expenses for category '" + category + "': $" + total);

}

// Method to save expenses to a file

**public** **void** saveExpensesToFile(String filename)

{

**try** (PrintWriter writer = **new** PrintWriter(**new** FileWriter(filename)))

{

**for** (Expense expense : expenses)

{

writer.println(expense.getDate() + "," + expense.getCategory() + "," + expense.getAmount());

}

System.***out***.println("Expenses saved to file: " + filename);

} **catch** (IOException e) {

System.***out***.println("Error saving expenses to file: " + e.getMessage());

}

}

// Method to load expenses from a file

**public** **void** loadExpensesFromFile(String filename)

{

**try** (Scanner fileScanner = **new** Scanner(**new** File(filename)))

{

expenses.clear();

**while** (fileScanner.hasNextLine())

{

String line = fileScanner.nextLine();

String[] parts = line.split(",");

String date = parts[0];

String category = parts[1];

**double** amount = Double.*parseDouble*(parts[2]);

Expense expense = **new** Expense(date, category, amount);

expenses.add(expense);

}

System.***out***.println("Expenses loaded from file: " + filename);

} **catch** (FileNotFoundException e) {

System.***out***.println("File not found: " + e.getMessage());

}

}

// Main method to run the application

**public** **static** **void** main(String[] args)

{

User expenseTracker = **new** User();

Scanner scanner = **new** Scanner(System.***in***);

**boolean** exit = **false**;

**while** (!exit)

{

System.***out***.println("\nExpense Tracker Menu:");

System.***out***.println("1. Add Expense");

System.***out***.println("2. Display All Expenses");

System.***out***.println("3. Calculate Category Total");

System.***out***.println("4. Save Expenses to File");

System.***out***.println("5. Load Expenses from File");

System.***out***.println("6. Exit");

System.***out***.print("\nEnter your choice: ");

**int** choice = scanner.nextInt();

**switch** (choice)

{

**case** 1:

expenseTracker.addExpense();

**break**;

**case** 2:

expenseTracker.displayExpenses();

**break**;

**case** 3:

System.***out***.print("Enter category to calculate total: ");

String category = scanner.next();

expenseTracker.calculateCategoryTotal(category);

**break**;

**case** 4:

System.***out***.print("Enter filename to save expenses: ");

String saveFile = scanner.next();

expenseTracker.saveExpensesToFile(saveFile);

**break**;

**case** 5:

System.***out***.print("Enter filename to load expenses: ");

String loadFile = scanner.next();

expenseTracker.loadExpensesFromFile(loadFile);

**break**;

**case** 6:

exit = **true**;

System.***out***.println("Exiting...");

**break**;

**default**:

System.***out***.println("Invalid choice. Please enter a number between 1 and 6.");

}

}

scanner.close();

}

}

**Output:**

Expense Tracker Menu:

1. Add Expense

2. Display All Expenses

3. Calculate Category Total

4. Save Expenses to File

5. Load Expenses from File

6. Exit

Enter your choice: