**User.java: Handles user information.**

java

Copy code

package com.expensetracker;

public class User {

private String username;

private String password;

public User(String username, String password) {

this.username = username;

this.password = password;

}

// Getters and Setters

public String getUsername() {

return username;

}

public String getPassword() {

return password;

}

}

**Expense.java: Represents an expense entry.**

java

Copy code

package com.expensetracker;

public 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;

}

// Getters and Setters

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;

}

}

**Step 4: Implement User Registration and Login**

**UserRegistration.java: Handles user registration and login.**

java

Copy code

package com.expensetracker;

import java.io.\*;

import java.util.Scanner;

public class UserRegistration {

private static final String USER\_FILE = "users.txt";

public static void registerUser() throws IOException {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter Username: ");

String username = scanner.nextLine();

System.out.print("Enter Password: ");

String password = scanner.nextLine();

try (BufferedWriter writer = new BufferedWriter(new FileWriter(USER\_FILE, true))) {

writer.write(username + "," + password);

writer.newLine();

}

System.out.println("User registered successfully!");

}

public static boolean loginUser(String username, String password) throws IOException {

try (BufferedReader reader = new BufferedReader(new FileReader(USER\_FILE))) {

String line;

while ((line = reader.readLine()) != null) {

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

if (userData[0].equals(username) && userData[1].equals(password)) {

return true;

}

}

}

return false;

}

}

**Step 5: Implement Expense Tracker Logic**

**ExpenseTracker.java: Main functionality for expense tracking.**

java

Copy code

package com.expensetracker;

import java.io.\*;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.Scanner;

public class ExpenseTracker {

private ArrayList<Expense> expenses = new ArrayList<>();

private static final String EXPENSE\_FILE = "expenses.txt";

public void addExpense() {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter date (YYYY-MM-DD): ");

String date = scanner.nextLine();

System.out.print("Enter category: ");

String category = scanner.nextLine();

System.out.print("Enter amount: ");

double amount = scanner.nextDouble();

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

expenses.add(expense);

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

}

public void listExpenses() {

if (expenses.isEmpty()) {

System.out.println("No expenses recorded.");

} else {

for (Expense expense : expenses) {

System.out.println(expense);

}

}

}

public void calculateCategoryWiseSum() {

HashMap<String, Double> categorySums = new HashMap<>();

for (Expense expense : expenses) {

categorySums.put(expense.getCategory(),

categorySums.getOrDefault(expense.getCategory(), 0.0) + expense.getAmount());

}

System.out.println("Category-wise Expense Summary:");

for (String category : categorySums.keySet()) {

System.out.println(category + ": " + categorySums.get(category));

}

}

public void saveExpensesToFile() throws IOException {

try (BufferedWriter writer = new BufferedWriter(new FileWriter(EXPENSE\_FILE))) {

for (Expense expense : expenses) {

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

writer.newLine();

}

}

System.out.println("Expenses saved to file.");

}

public void loadExpensesFromFile() throws IOException {

try (BufferedReader reader = new BufferedReader(new FileReader(EXPENSE\_FILE))) {

String line;

while ((line = reader.readLine()) != null) {

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

Expense expense = new Expense(data[0], data[1], Double.parseDouble(data[2]));

expenses.add(expense);

}

}

System.out.println("Expenses loaded from file.");

}

}

**Step 6: Main Method to Run the Application**

**Main.java: Entry point for the program.**

java

Copy code

package com.expensetracker;

import java.io.IOException;

import java.util.Scanner;

public class Main {

public static void main(String[] args) throws IOException {

Scanner scanner = new Scanner(System.in);

ExpenseTracker expenseTracker = new ExpenseTracker();

UserRegistration userRegistration = new UserRegistration();

// User Registration

System.out.println("Welcome to the Expense Tracker");

System.out.println("1. Register\n2. Login");

int choice = scanner.nextInt();

scanner.nextLine(); // Consume newline

if (choice == 1) {

userRegistration.registerUser();

}

// User Login

System.out.print("Enter Username: ");

String username = scanner.nextLine();

System.out.print("Enter Password: ");

String password = scanner.nextLine();

if (userRegistration.loginUser(username, password)) {

System.out.println("Login successful!");

// Load expenses from file

expenseTracker.loadExpensesFromFile();

// Expense Tracker Menu

int option;

do {

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

System.out.println("2. List Expenses");

System.out.println("3. Calculate Category-wise Sum");

System.out.println("4. Save Expenses");

System.out.println("5. Exit");

option = scanner.nextInt();

switch (option) {

case 1:

expenseTracker.addExpense();

break;

case 2:

expenseTracker.listExpenses();

break;

case 3:

expenseTracker.calculateCategoryWiseSum();

break;

case 4:

expenseTracker.saveExpensesToFile();

break;

case 5:

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

break;

default:

System.out.println("Invalid option. Try again.");

}

} while (option != 5);

} else {

System.out.println("Invalid credentials!");

}

}

}