Logan DeWeese

10/6/2023

SDEV 140  
  
Final Project

Expense Tracker Lite

Github URL: https://github.com/ldeweese2/DeWeeseLoganFinalProject/tree/main

This Python application is a simple Expense Tracker Lite built using the Tkinter library for creating a graphical user interface. It helps users track their expenses by entering the expense amount and category. Here's a brief explanation of its features:

1. **Expense Entry**: Users can input their expenses by specifying the expense amount and category through the provided input fields.
2. **Add Expense**: By clicking the "Add Expense" button, the application validates the input, ensuring that both fields are filled and that the expense amount is a valid number. If the input is valid, the expense is added to the list.
3. **View Summary**: Users can click the "View Summary" button to see an overview of their expenses. It opens a new window displaying a list of expenses grouped by category, along with the total expenses.
4. **Exit**: The "Exit" button allows users to close the application.
5. **Images**: The application also features two images displayed side by side, adding a visual element to the interface.

This Expense Tracker Lite provides a simple and user-friendly way to keep track of expenses and view a summary of spending. It's a handy tool for individuals looking to manage their finances effectively.

**Test Data 1: Valid Input**

* Expense: 100.50
* Category: Groceries

**Test Data 2: Missing Expense**

* Expense: (empty)
* Category: Groceries

**Test Data 3: Missing Category**

* Expense: 75.00
* Category: (empty)

**Test Data 4: Invalid Expense**

* Expense: abc
* Category: Utilities

**Test Data 5: Multiple Entries**

* Expense 1: 50.00, Category: Dining
* Expense 2: 30.25, Category: Entertainment
* Expense 3: 120.75, Category: Shopping

**Test Data 6: Viewing Summary**

* Add a few valid expenses and then click "View Summary."

**Results and Fixes:**

1. **Test Data 1 (Valid Input)**:
   * Expected: Expense added successfully.
   * Actual: Expense added successfully.
2. **Test Data 2 (Missing Expense)**:
   * Expected: An error message should display, indicating that both fields are required.
   * Actual: Error message displayed correctly.
3. **Test Data 3 (Missing Category)**:
   * Expected: An error message should display, indicating that both fields are required.
   * Actual: Error message displayed correctly.
4. **Test Data 4 (Invalid Expense)**:
   * Expected: An error message should display, indicating that the expense must be a valid number.
   * Actual: Error message displayed correctly.
5. **Test Data 5 (Multiple Entries)**:
   * Expected: Multiple expenses should be added successfully, and the summary should reflect the correct total.
   * Actual: Multiple expenses added correctly, and the summary displayed the total correctly.
6. **Test Data 6 (Viewing Summary)**:
   * Expected: The summary window should display a list of expenses with categories and the total expenses.
   * Actual: The summary window displayed correctly with the list of expenses and total.