

## Programming Assignment: Expense Tracker Application

**Task:** You are tasked with creating an Expense Tracker application that allows users to manage their expenses. The application should involve reading and writing from a file, obtaining input from the user, performing simple computations, displaying information based on user input and calculations, and formatting the output for readability.

### Instructions:

- 1) Create a text file named "expenses.txt" to store the expense data. Each line in the file should contain an expense record in the format: "Category,Amount".
- 2) Write a program in python that accomplishes the following:
  - a) Reads the contents of the "expenses.txt" file.
  - b) Displays a menu to the user with the following options:
    - i) Add an expense
    - ii) View expenses by category
    - iii) Calculate total expenses
    - iv) Exit c.
  - c) Based on the user's choice, perform the appropriate actions:
    - i) If the user chooses to add an expense, prompt for the expense category and amount, and append the record to the "expenses.txt" file.
    - ii) If the user chooses to view expenses by category, prompt for a category name and display all expenses for that category.
    - iii) If the user chooses to calculate total expenses, compute and display the total expenses across all categories.
    - iv) If the user chooses to exit, terminate the program.
  - d. Format the output for readability, such as aligning columns and adding appropriate headers.

### Requirements:

1. Your program should handle any file-related errors appropriately (e.g., file not found, read/write errors).
2. Validate user input to ensure it is within the acceptable range or format.
3. Perform computations, such as calculating the total expenses or filtering expenses by category, as required by the user's choice.
4. Display the information in a formatted manner, such as using tables or a similar structure for better readability.
5. Maintain the state of the expense data across multiple program runs by reading from and writing to the "expenses.txt" file.

### Example Output:

Expense Tracker Application

Menu:

1. Add an expense
2. View expenses by category

3. Calculate total expenses
4. Exit

Enter your choice: 1

Add an Expense

Expense Category: Grocery Expense Amount: \$50.00 Expense added successfully!

Menu:

1. Add an expense
2. View expenses by category
3. Calculate total expenses
4. Exit

Enter your choice: 2

View Expenses by Category

Enter the category name: Grocery

Expenses for Category: Grocery

Category	Amount
----------	--------

\*\*\*\*\*

Grocery	\$50.00
---------	---------

Menu:

1. Add an expense
2. View expenses by category
3. Calculate total expenses
4. Exit

Enter your choice: 3

Total Expenses: \$50.00

Menu:

1. Add an expense
2. View expenses by category
3. Calculate total expenses
4. Exit

Enter your choice: 4

Exiting the Expense Tracker Application...

Note: The above output is just an example. Feel free to format the output and menu options in any way that you think makes it clear and readable.

**Submission:** Submit your program's source code along with any necessary files, such as "expenses.txt" on Gradescope. Provide clear instructions on how to run the program, including any dependencies or setup required.