**Expense Tracker – Function Overview**

**1.**

**ExpINput()**

Purpose: Collects new expense entries from the user.

* Asks the user how many expenses to add.
* For each expense, prompts for:
  + date (YYYY-MM-DD)
  + category (Food or Travel)
  + amount (converted to float)
  + Description
* Returns a list of expense dictionaries.

Notes:

* Handles invalid input using try/except.
* Each expense is stored as a dictionary with keys: date, category, amount, Description.

**2.**

**calculate\_expenses(exp, bud)**

Purpose: Computes the total expenses and remaining budget.

How it works:

* Converts each expense amount to float (if needed).
* Sums all amounts using sum() or a lambda.
* Subtracts total expenses from the user’s budget bud.
* Prints either remaining budget or a warning if over budget.

Input:

* exp → list of expense dictionaries
* bud → float (user’s budget)

**3.**

**Print\_expense(exps)**

Purpose: Displays all expense records.

How it works:

* Loops through the list of dictionaries.
* Prints each dictionary (or could be formatted for readability).

**4.**

**Save\_expenses(exps)**

Purpose: Saves expense records to a CSV file.

How it works:

* Opens expenses.csv in write mode.
* Uses csv.DictWriter to write header and rows.
* Writes all expense dictionaries to the CSV.

**5.**

**Load\_expenses()**

Purpose: Loads expense records from a CSV file.

How it works:

* Opens expenses.csv in read mode.
* Uses csv.DictReader to read each row as a dictionary.
* Returns a list of dictionaries.

Notes:

* CSV fields are read as strings, so numeric fields may need float() conversion.

**6.**

**Budget()**

Purpose: Prompts user to input their monthly budget.

How it works:

* Loops until user enters a valid number (float).
* Returns the budget amount.

**7.**

**Interactive(exp)**

Purpose: Main menu and program loop.

How it works:

* Provides options:
  1. Add Expenses
  2. View Expenses
  3. Track Budget
  4. Save Expenses
  5. Exit
* Calls the corresponding functions based on user choice.
* Maintains exp as a flat list of dictionaries.

**8.**

**main()**

Purpose: Entry point of the program.

How it works:

* Initializes an empty list exp.
* Calls Interactive(exp) to start the menu loop.

 High-Level Flow of Program:

1. Start → main()
2. Show menu → Interactive()
3. User selects action → calls functions like ExpINput(), Print\_expense(), etc.
4. Expenses can be saved/loaded from CSV.
5. Budget tracking uses calculate\_expenses()