Wallet (incomes and expenses) Documentation

About:

This project is developed based on C programming language using Linked Lists and File handling.

Main Purpose:

Managing and organizing the incomes and expenses for the daily use wallet.

Allowing the user to keep up with his incomes and expenses.

Knowing the history details of his transactions.

Structures:

1- Structure for the date with the format (yyyy/mm/dd)

2- Structure for the single input ”node” which contains members: date, category, amount, and pointer to the next node

3- Structure for the history of transactions with two pointers pointing to the income and expense.

Functions Used:

1- Insert():

This function takes the date, category, amount, and type from the user and create a new transaction.

We can use it to add a new income or new expense.

It also allows the user to create a new category if it doesn’t exist already.

Return value: void

2- Show():

This function displays all the transactions of the user with date, category, amount and currency depending on the choice of the user which is one of the two choices: income or expense.

Return value: void

3- **Save_income():**

This function takes the data from the user with the help of insert() function and save it in the file income to not lose the data after closing the program.

Return value: void

4- **Save_expense():**

This function takes the data from the user with the help of insert() function and save it in the file expense to not lose the data after closing the program.

Return value: void

5- **Read_income()**

This function Load the data from the file income that we have created previously with the help of Save_income() function.

Return value: address of required data

6- **Read_expense()**

This function Load the data from the file expense that we have created previously with the help of Save_expense() function.

Return value: address of required data

7- **Delete_history()**

This function is used to free the memory whenever we call it

Return value: void

8- **Save_history()**

This function is used to update the value of current income and current expense and save them in history file before terminating the program.

Return value: void

9- **Read_history():**

This function is used to get the latest data of current income and current expense from the history file that we have saved before terminating the program by the help of the Save_history() function.

Return value: address of the required data.

Data Structure and Algorithms used:

Linked List

Pointers

File handling

Structures

Main Function:

Using Loops and Switch case, we were able to make a menu and interface for the user to make it easy for him to choose between the options.

Like the following:

Choose an Option from The List Below:

1. Add Income
2. Add Expense
3. Show Income History
4. Show Expense History
5. Exit