

Backend Design Doc

Overview

Our backend reads in both the master accounts file and the merged transactions summary file and appends them to a local dictionary and list respectively to allow for simplicity of reading and updating the information. Every line in the transactions list is processed by reading and calling each function specified in the file (create/delete account, transfer, deposit and withdraw). Each function edits the accounts dictionary according to their functionality. After every transaction has been processed, the master accounts file is updated and the valid accounts list (used by the frontend of the program) is written.

Global Variables

accounts - Dictionary in the format "account number": ("balance", "account name") read from the master accounts file

transactions - List with each line a string read in from the merged transactions summary file ("CCC AAAA MMMM BBBB NNNN")

accounts_filename - name of the master accounts file

transactions_filename - name of the merged transactions summary file

Methods

read_accounts()

Opens master accounts file within a try;except block and adds each line to the *accounts* global variable for simple readability within the rest of the program.

read_transactions()

Opens merged transactions file within a try;except block and adds each line to the *transactions* global variable for simple readability within the rest of the program.

process_transactions()

Loops through all the transactions from the transactions file and calls each respective function based on the command given (deposit, withdraw, transfer, create account, delete account). After all transactions are created, writes the updated information to the *valid_accounts_list* file.

create_account(accountNumber, accountName)

Creates account by adding account number and account name read from the merged transaction file into the global *accounts* variable.

delete_account(accountNumber, accountName)

Deletes account by removing account corresponding to the account number specified in the merged transactions file from the global *accounts* variable.

transfer(fromAccount, toAccount, money)

Adds amount specified to the specified *toAccount* in the merged transactions file to the global *account* variable as well as subtracts the same amount from the specified *fromAccount*.

deposit(accountNum, amount)

Adds specified amount to the current balance to the account with the specified account number in the *account* global variable.

withdraw(accountNum, amount)

Subtracts specified amount to the current balance to the account with the specified account number in the *account* global variable.

main()

Controls overall run of the program by setting the *accounts_filename* and the *transactions_filename* and calling *read_accounts()*, *read_transactions* and *process_transactions()*.