**INTRODUCTION**

The system was developed for only one user with the username “Fatin” and the password “meowmeow” which was stored in a text file. It consists of five modules; Login, Assets, Income, Expenses, Reporting. In every module (Login excluded), user is able to add, edit, remove and view components of each asset.

The system is designed to manage the user’s finance by enabling the user to record the daily, weekly and monthly expenditure of the user. The system is developed using NetBeans IDE 8.0.2 with Java programming language.

**CLASS DESCRIPTION**

Login Module

The login module allows the user to input username and password. As stated earlier, only one user is able to access this system. The algorithm for Login Module is quite simple. A text file containing a username and a password was created. The user inputs username and password and the compiler checks if what was entered matches the username and password inside the text file. If it does, then the system will display a “main menu” to choose from. Here, the user will be able to choose to something to Assets, Income, Expenses or Report.

Asset Module

This module is where the user’s list of property is stored. If the user choose Assets from the main menu, a new menu is displayed. User will choose either to view, add, remove or edit the list. If user choose to view, the system will simply open a text file containing the list, and display it for the user. If user choose to add, then the system will prompt user to enter the asset category (eg: house, car, etc) and the value of the asset. These inputs will be added into the list. If user choose to edit, the system will prompt user to choose which line/asset to edit. After user enter the required input, system will process the request and override the text file with the changes made. If the user choose to delete assets, the system will prompt user to enter the line/asset that the user wish to delete. After a value is entered, the system will remove the selected line/asset from the text file.

Income Module

Income module allows user to add or view income. Upon choosing to open income module from the main menu, user is presented with options of whether to add or view income. If user choose add, system will prompt user to choose a category (eg: Salary, Allowance, etc) and then after choosing a category, user is required to enter amount of income. This input will be stored into a text file that was created especially for this specific module. Meanwhile, if user choose view, system will open the aforementioned text file and display it to user. Here user can view the monthly income.

Expenses Module

Expenses module is where user inputs and view expenditures. After choosing Expenses module from the main menu, user is required to choose what action to take. It is either to view or to add expense into the list. If user choose view, system simply reads a text file created specifically for this class to display the list of expenses. However, if user choose to add expense, system will prompt user to choose category (eg: food, electronics, etc…) and then to enter the amount spent. This module is to keep track of the user’s expenditure and thus aids in saving money.

Report Module

Report module is where user really see the moeny flow. When user view the financial report, system will read a text file that has initially been created for this class. Reports are made in three formats: daily, weekly, monthly. User will input daily expenses, which will be accumulated weekly. System will then sum up all total income together and then sum up total expenses together. To calculate whether or not user had loss or gain in a particular month, a calculation is made by subtracting total expenses with total income. If balance is negative, user had a loss and if balance is positive, user had a gain.

**FLOW CHART**

START

READ:

**String** username, password

Y

N

TRY AGAIN?

CORRECT?

N

DISPLAY: Main menu

Y

READ: int users\_action



A

CASE 1



N

B

CASE 2



C

N

CASE 3

N



D

CASE 4

N

END

A

ASSET MODULE

READ: int userAction userAction

CASE 1

VIEW

CASE 2

ADD

CASE 3

EDIT

REMOVE

CASE 4

END

B

INCOME MODULE

READ: int userAction userAction

CASE 1

VIEW

CASE 2

ADD

CASE 3

EDIT

REMOVE

CASE 4

END

C

EXPENSE MODULE

READ: int userAction userAction

CASE 1

VIEW

CASE 2

ADD

END

D

REPORT MODULE

READ: int userAction userAction

DAILY REPORT (INPUT BY USER IN EXPENSE MODULE)

WEEKLY REPORT

MONTHLY REPORT

X=totalIncome-totalExpense

Y

Is x negative?

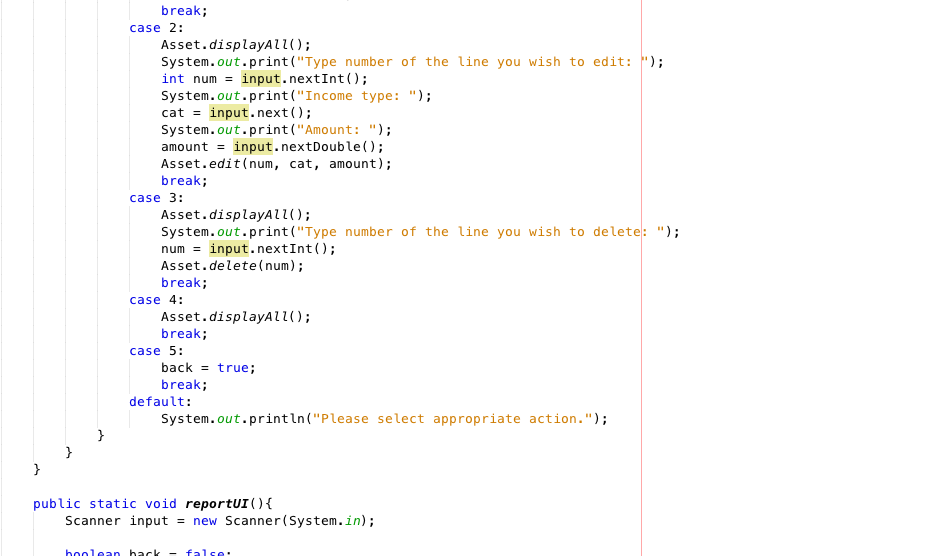
LOSS

N

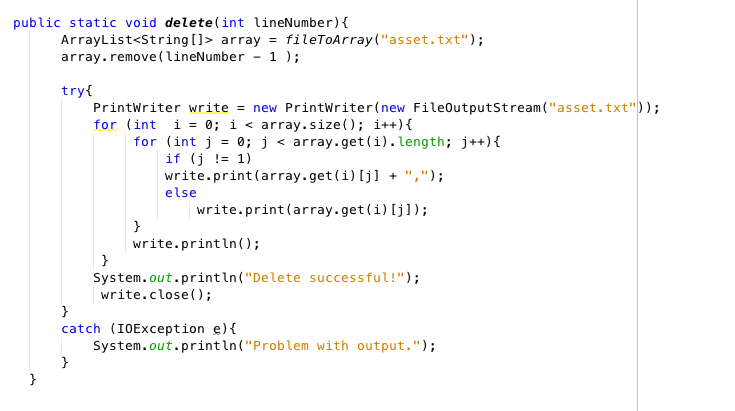
GAIN

END

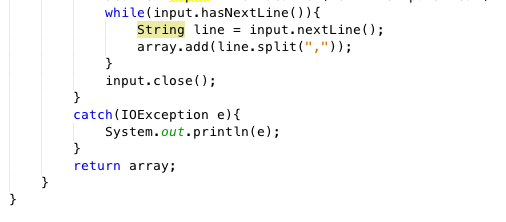
**EXTRACTS**



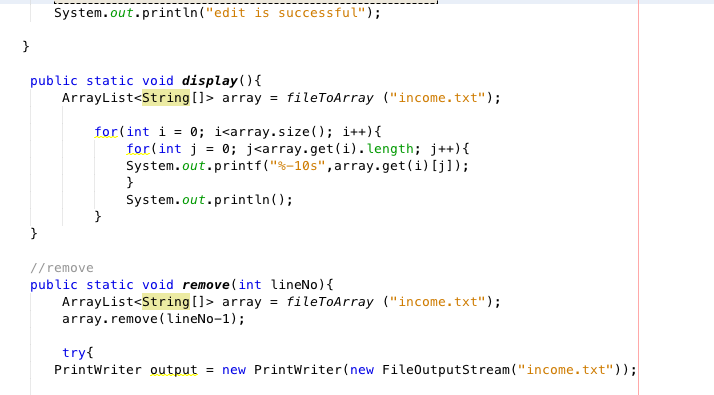
Part of the codes for main method



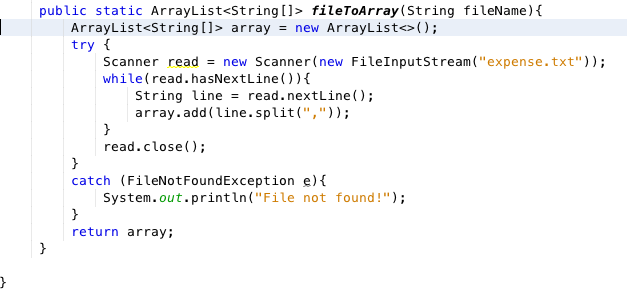
Part of the codes for asset class



Part of the codes for income class



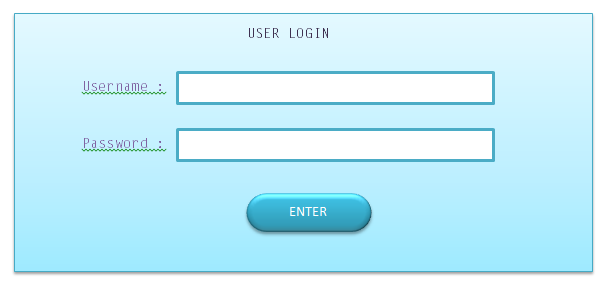
Part of the codes for income class



Part of the codes for expense class



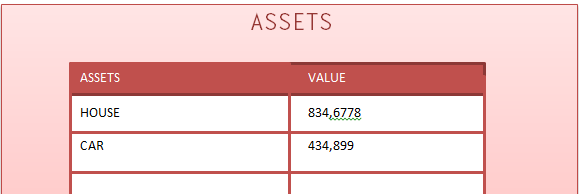
Part of the codes for report class



User login interface



User income interface



User asset interface