This program is intended to teach students the concepts of fiscal responsibility via an interactive system. This application will demonstrate the power of compound interests. When the user inputs Initial investment amount, monthly deposit, annual interest (compounded) and a Number of Years two static reports will display. The first report will indicate the year-end balances and interest earned if no additional monthly deposits are made. The second report will indicate the year-end balances and year-end interest earned based on monthly deposit value that was input by the user.

class FinancialSimulation {

Declarations:

Private num initialInvestment

Private num monthlyDeposit

Private int numberYears

Private int InterestAnnual

public void Function requestDataInput() {

Display the user input menu requesting the variables

Initial Investment Amount: $

Monthly Deposit: $

Annual Interest: %

Number of Years:

Assign those variable inputs to the public variables of class,

Calculate numberMonths, assign

Calculate interestMonthly, assign

}

public num Function calculateCompoundInterestNoDeposit(num calcYear) {

Calculate the compound interest earned without monthly deposit

Return compound interest earned for col 1

}

Public num Function calculateAccountBalanceNoDeposit(num calcYear){

Calculate the year end account balance for col 2

}

public num Function calculateCompoundInterestDeposit(num initialInvestment, num monthlyDeposit, num interestRate, num numberOfYears) {

Calculate the compound interest earned with monthly deposit

Return compound interest earned for col 1

}

Public num Function calculateAccountBalanceDeposit(num calcYear){

Calculate the year end account balance with deposit for col 2

}

Public num

public void Function printTables() {

num balance (stores the bank balance at start of year)

num closingBalance (stores the bank balance at end of year)

Output “Balance and Interest Without Additional Monthly Deposits”

Output Columns “Year” , “Year End Balance”, “Year End Earned Interest”

Loop each month using calculateCompoundInterestNoDeposit() & calculateAccountBalanceNoDeposit() functions

Output Table Rows

Output “Balance and Interest With Additional Monthly Deposits”

Output Columns “Year” , “Year End Balance”, “Year End Earned Interest”

Loop each month using calculateCompoundInterest() & calculateAccountBalance() functions

Output Table Rows

}

public void Function printMenu() {

Print a menu providing options to run another simulation or exit with 6

}

public void SetInitialInvestment(){

Set initial investment with user input

}

Public num GetInitialInvestment(){

Return value of Initial Investment

}

public void SetMonthlyDeposit(){

Set Monthly Deposit to user input

}

Public num GetMonthlyDeposit(){

Return the Monthly Deposit Value

}

public void SetNumberYears(){

Set number of years to user input

}

Public num GetNumberYears(){

Return number of Years

}

Public void SetInterestAnnual(){

Set Interest Annual based upon user input

}

Public num GetInterestAnnual(){

Return Annual Interest

}

end class

In the Main Function {

Int userInput;

Int simulationNumber

Create an instance of Financial Simulation Class

requestUserInput()

printTables()

Initialize User Menu Loop

While Menuinput does not equal 6 {

IF 0 printMenu

IF 1 Change Initial Investment

IF 2 Change Monthly Deposit

IF 3 Change Number of Years

IF 4 Change Annual Interest

IF 5 Print Tables

IF 6 Exit Program

}