/\*\*

\* Deborah Barndt

\* 9-22-16

\* Client.java

\* Lab 2

\* This program will create an abstract class Client to allow for three abstract methods the bank needs to process.

\* Written by Deborah Barndt. \*/

**package** bankofiit;

**import** java.net.MalformedURLException;

**import** java.util.ArrayList;

**import** java.util.List;

/\*\*

\* **@author** Deborah Barndt

\*

\*/

**public** **abstract** **class** Client

{

/\*\*

\*

\*/

**public** Client()

{

// **TODO** Auto-generated constructor stub

}

**private** String id, // the client's id

sex, // value of whether the client is male or female

region, // value of the client's city, town, rural, suburban

married, // value of whether the client is married

car, // value of whether the client has a car

save\_act, // value of whether the client has a savings account

current\_act, // value of whether the client has a current account

mortgage, // value of whether the client has a mortgage

pep; // value of whether the client is a politically exposed person

**private** **int** age, // value of the client's age

children; // amount of children the client has

**private** **double** income; // amount of income the client has

//Create the bank records object to be processed.

BankRecords clientInfo[];

List<List<String>> bankInfo = **new** ArrayList<List<String>>();

// Setter to assign the clientInfo to the ArrayList.

**public** **void** setclientInfo(BankRecords[] clientInfo)

{

**this**.clientInfo = clientInfo;

}

// Setter to assign the id of the client.

**public** **void** setId(String id)

{

**this**.id = id;

}

// Setter to assign the age of the client.

**public** **void** setAge(**int** age)

{

**this**.age = age;

}

// Setter to assign the sex of the client.

**public** **void** setSex(String sex)

{

**this**.sex = sex;

}

// Setter to assign the region of the client.

**public** **void** setRegion(String region)

{

**this**.region = region;

}

// Setter to assign the income of the client.

**public** **void** setIncome(**double** income)

{

**this**.income = income;

}

// Setter to assign the married status of the client.

**public** **void** setMarried(String married)

{

**this**.married = married;

}

// Setter to assign the amount of children of the client.

**public** **void** setChildren(**int** children)

{

**this**.children = children;

}

// Setter to assign the car status of the client.

**public** **void** setCar(String car)

{

**this**.car = car;

}

// Setter to assign the savings account status of the client.

**public** **void** setSave\_act(String save\_act)

{

**this**.save\_act = save\_act;

}

// Setter to assign the current account status of the client.

**public** **void** setCurrent\_act(String current\_act)

{

**this**.current\_act = current\_act;

}

// Setter to assign the mortgage status of the client.

**public** **void** setMortgage(String mortgage)

{

**this**.mortgage = mortgage;

}

// Setter to assign the politically exposed person status of the client.

**public** **void** setPep(String pep)

{

**this**.pep = pep;

}

// Getter to get the clientInfo from the ArrayList

//public BankRecords[] getclientInfo()

**public** BankRecords[] getclientInfo()

{

**return** clientInfo;

}

// Getter to get the id of the client.

**public** String getId()

{

**return** id;

}

// Getter to get the age of the client

**public** **int** getAge()

{

**return** age;

}

// Getter to get the sex of the client.

**public** String getSex()

{

**return** sex;

}

// Getter to get the region of the client.

**public** String getRegion()

{

**return** region;

}

// Getter to get the income of the client.

**public** **double** getIncome()

{

**return** income;

}

// Getter to get the married status of the client.

**public** String getMarried()

{

**return** married;

}

// Getter to get the amount of children of the client.

**public** **int** getChildren()

{

**return** children;

}

// Getter to get the car status of the client.

**public** String getCar()

{

**return** car;

}

// Getter to get the savings account status of the client.

**public** String getSave\_act()

{

**return** save\_act;

}

// Getter to get the current account status of the client.

**public** String getCurrent\_act()

{

**return** current\_act;

}

// Getter to get the mortgage of the client.

**public** String getMortgage()

{

**return** mortgage;

}

// Getter to get the politically exposed person status of the client.

**public** String getPep()

{

**return** pep;

}

// Method to read all the record data from the CSV file in the ArrayList.

**abstract** **void** readData();

// Method to process all the record data from the ArrayList and add the data into each of the instance fields.

**abstract** **void** processData();

// Method to print the first 25 records for various fields to be displayed.

**abstract** **void** printData() **throws** MalformedURLException;

}

/\*\*

\* Deborah Barndt

\* 9-22-16

\* BankRecords.java

\* Lab 2

\* This program will utilize the Client abstract methods and generate the client records from the csv file.

\* It will parse and process bank data from the file.

\* Written by Deborah Barndt. \*/

**package** bankofiit;

**import** java.awt.Color;

**import** java.io.BufferedReader;

**import** java.io.FileNotFoundException;

**import** java.io.FileReader;

**import** java.io.IOException;

**import** java.net.MalformedURLException;

**import** java.net.URL;

**import** java.text.DecimalFormat;

**import** java.text.SimpleDateFormat;

**import** java.util.ArrayList;

**import** java.util.Arrays;

**import** java.util.Calendar;

**import** java.util.List;

**import** javax.swing.ImageIcon;

**import** javax.swing.JOptionPane;

**import** javax.swing.JTextArea;

**import** javax.swing.UIManager;

/\*\*

\* **@author** Deborah Barndt

\*

\*/

**public** **class** BankRecords **extends** Client

{

// Print a time stamp for program.

String timeStamp = **new** SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(Calendar.*getInstance*().getTime());

@Override

// Method to read all the record data from the bank-Detail.csv file.

**void** readData()

{

String line = " ";

// Use the try with resources statement for file IO.

**try**(BufferedReader detail = **new** BufferedReader(**new** FileReader("bank-Detail.csv")))

{

**int** index = 0;

**while**((line = detail.readLine()) != **null**)

{

// Read from the bank-Detail.csv file.

bankInfo.add(Arrays.*asList*(line.split(",")));

//System.out.println(bankInfo.get(index++));

//index++;

}

}

// Catch in case the file is not found.

**catch** (FileNotFoundException e)

{

e.printStackTrace();

}

// Catch in case an input/output in not found.

**catch** (IOException e)

{

e.printStackTrace();

}

processData();

}

@Override

// Method to process all the record data from the bank-Detail.csv file.

**void** processData()

{

// Process the data from ArrayList into an array.

// Use enhanced for loop for (data type elementVariable: array)

**int** idx = 0;

clientInfo = **new** BankRecords[bankInfo.size()];

// Enhanced for loop to set the data in the ArrayList.

**for** (List<String> rowData : bankInfo)

{

// Use setters to parse out your data and convert the data types

clientInfo[idx] = **new** BankRecords();

clientInfo[idx].setId(rowData.get(0));

clientInfo[idx].setAge(Integer.*parseInt*(rowData.get(1)));

clientInfo[idx].setSex(rowData.get(2));

clientInfo[idx].setRegion(rowData.get(3));

clientInfo[idx].setIncome(Double.*parseDouble*(rowData.get(4)));

clientInfo[idx].setChildren(Integer.*parseInt*(rowData.get(6)));

clientInfo[idx].setCar(rowData.get(7));

clientInfo[idx].setSave\_act(rowData.get(8));

clientInfo[idx].setCurrent\_act(rowData.get(9));

clientInfo[idx].setMortgage(rowData.get(10));

clientInfo[idx].setPep(rowData.get(11));

idx++;

}

}

@Override

// Method to print the first 25 records from the bank-Detail.csv file.

**void** printData() **throws** MalformedURLException

{

// Create a Text Area

JTextArea outputArea = **new** JTextArea (30, 48);

// Print the data from the ArrayList

outputArea.append(" ID\t" + " AGE\t" + " SEX\t" + " REGION\t" + " INCOME\t" + " MORTGAGE" + "\n\n");

// For loop to print the first 25 record from the ArrayList.

**for** (**int** idx = 0; idx <= 24; idx++)

{

// Use getters to get the data

outputArea.append(" " + clientInfo[idx].getId() + "\t");

outputArea.append(" " + clientInfo[idx].getAge() + "\t");

outputArea.append(" " + clientInfo[idx].getSex() + "\t");

outputArea.append(" " + clientInfo[idx].getRegion() + "\t");

outputArea.append(" " + clientInfo[idx].getIncome() + "\t");

//outputArea.append(" " + clientInfo[idx].getMarried() + "\t");

//outputArea.append(" " + clientInfo[idx].getChildren() + "\t");

//outputArea.append(" " + clientInfo[idx].getCar() + "\t");

//outputArea.append(" " + clientInfo[idx].getSave\_act() + "\t");

//outputArea.append(" " + clientInfo[idx].getCurrent\_act() + "\t");

outputArea.append(" " + clientInfo[idx].getMortgage() + "\n");

//outputArea.append(" " + clientInfo[idx].getPep() + "\n");

}

// Program time stamp

outputArea.append("\n\n" + " Current date: " + timeStamp + "\n Programmed by Deborah Barndt\n");

// Change icon inside dialog box.

**final** ImageIcon icon = **new** ImageIcon(**new** URL("https://cdn1.snapappts.com/media/96x96xclient-documents.png.pagespeed.ic.rI7dpZo8Wa.png"));

// Output for the for loop to display the data.

// Display results in a dialog box.

JOptionPane.*showMessageDialog*(**null**, outputArea, "Bank of IIT Client Information",

JOptionPane.***INFORMATION\_MESSAGE***, icon);

}

**public** **static** **void** main(String[] args) **throws** MalformedURLException

{

// Change the color of the background of the dialog box

UIManager UI = **new** UIManager();

UI.*put* ("OptionPane.background", Color.***orange***);

UI.*put* ("Panel.background", Color.***orange***);

BankRecords client = **new** BankRecords();

client.readData();

client.printData();

}

}

Output of the printData Method in a dialog box.

