**PROJECT Bank record generations 100 points**

**Objective** To write a program that parses and processes bank data from a file.

**PROJECT DESCRIPTION**

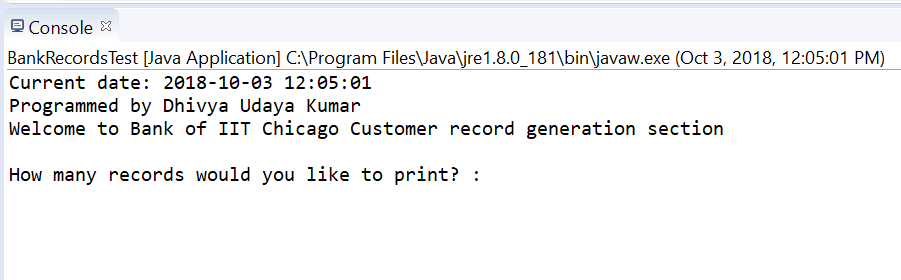
Bank of IIT has gotten their hands on some interesting data which will allow for possible loans to various clients from various regions.

Accompanying the labs specs is a csv (comma separated value) file named **bank-Detail.csv** which contains valuable raw data to allow the bank to process loans based on client details from the file.

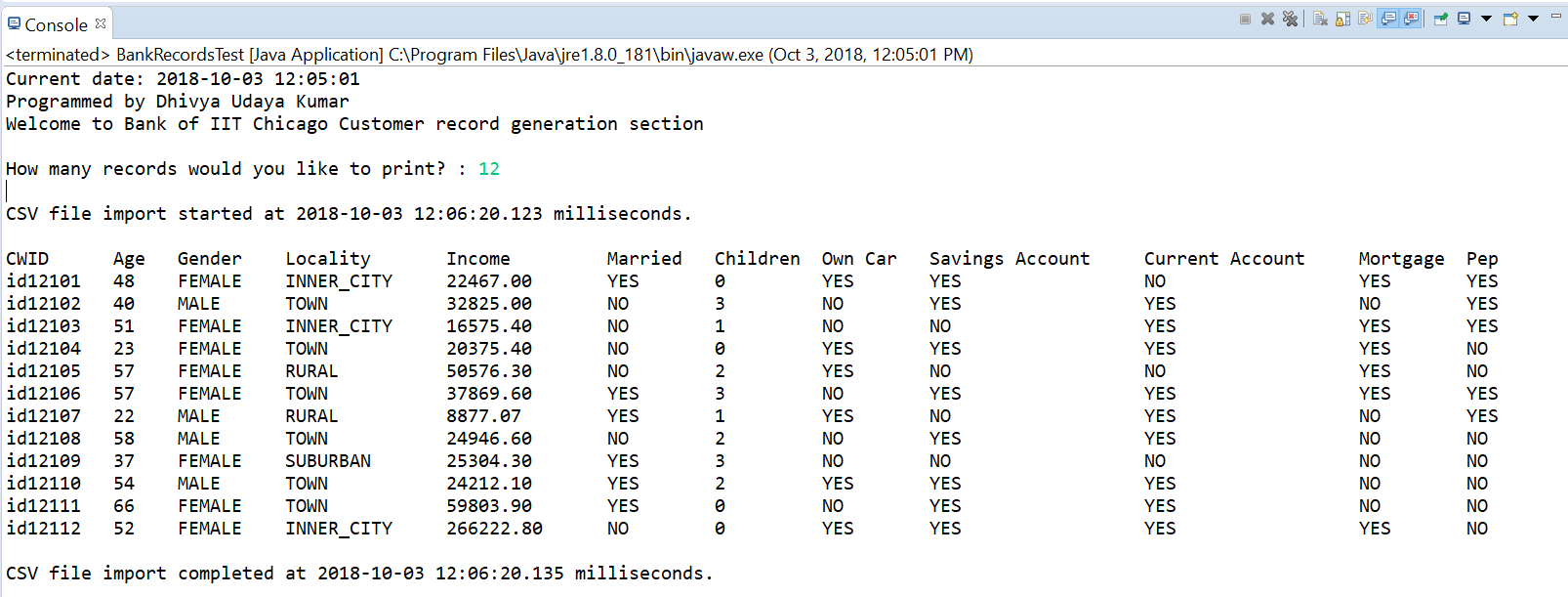
You need to parse the data and print record data for future loan considerations.

**Snapshots of Testcases:**

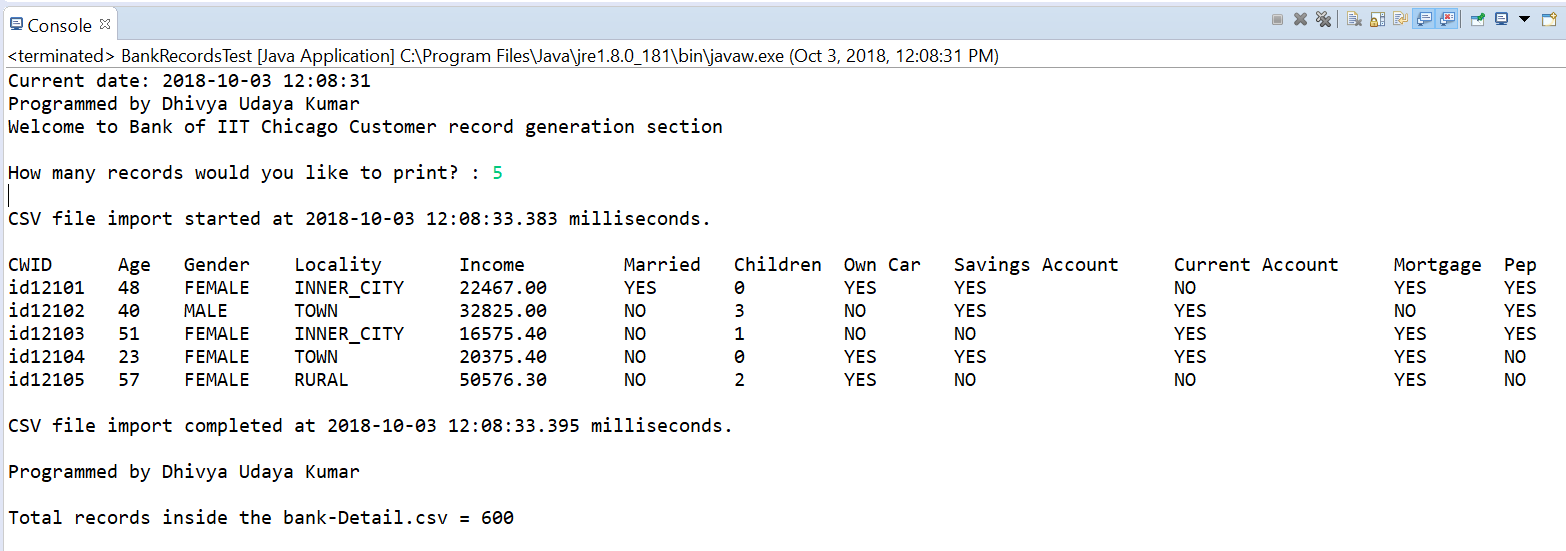
1. Initiation of BankRecordTest class



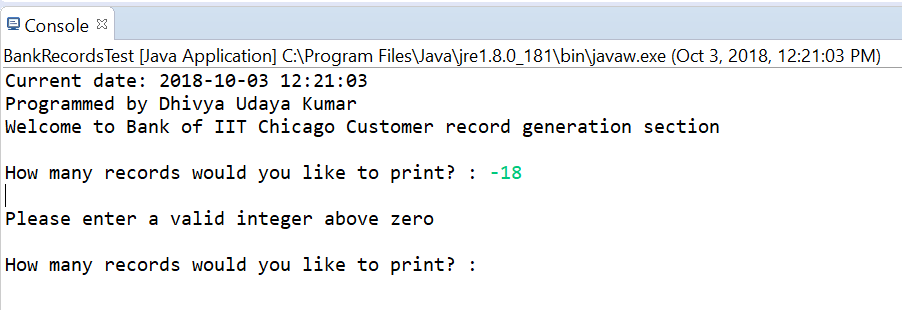
1. Display records based on user’s input for number of records



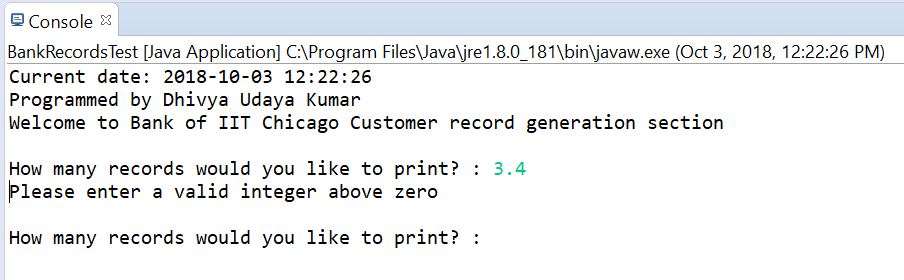
1. Display total records present in the csv file



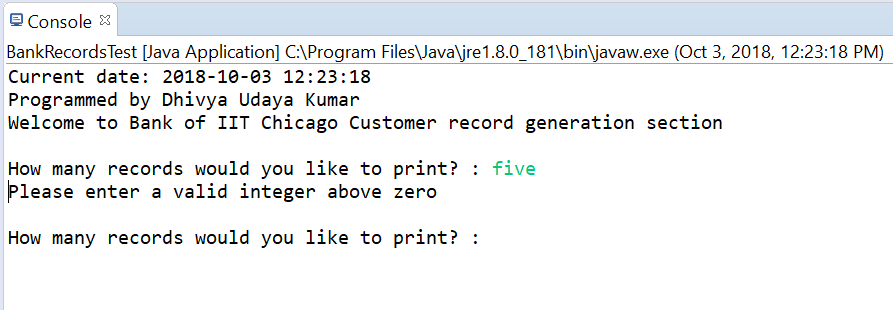
1. User’s input is negative



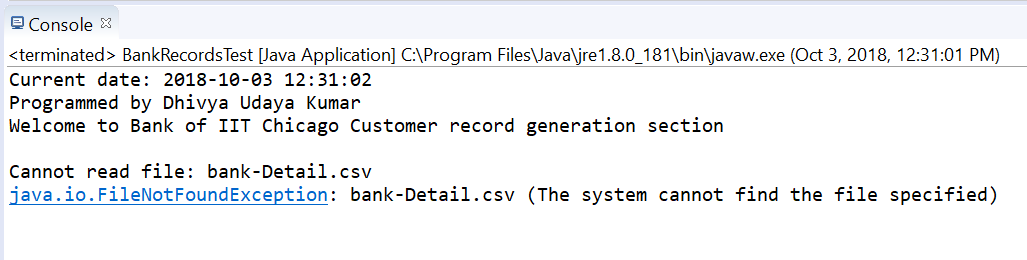
1. User’s input is not an integer



1. User’s input is not a number



1. Exception if file is not available



**Source Code:**

1. BankRecords.java

/\*

\* Class name: BankRecords

\* Description: Class contains methods to read, process and print data extracted from the CSV file

\* Programmed by: Dhivya Udaya Kumar

\* CWID: A20432502

\* Date: 3-Oct-2018

\*/

import java.io.BufferedReader;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.IOException;

import java.time.LocalDateTime;

import java.time.format.DateTimeFormatter;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

import java.util.Scanner;

public class BankRecords extends Client {//class-S

    // Static fields for IO processing

    // Array of BankRecords objects

    public static BankRecords robjs[] = new BankRecords[600];

    // arrayList to hold spreadsheet rows & columns

    static ArrayList<List<String>> array = new ArrayList<>();

    // Instance fields

    private String id;

    private int age;

    private String sex;

    private String region;

    private double income;

    private String married;

    private int children;

    private String car;

    private String save\_act;

    private String current\_act;

    private String mortgage;

    private String pep;

    // Getters and setters for the instance fields

    public String getId() {

        return id;

    }

    public void setId(String id) {

        this.id = id;

    }

    public int getAge() {

        return age;

    }

    public void setAge(int age) {

        this.age = age;

    }

    public String getSex() {

        return sex;

    }

    public void setSex(String sex) {

        this.sex = sex;

    }

    public String getRegion() {

        return region;

    }

    public void setRegion(String region) {

        this.region = region;

    }

    public double getIncome() {

        return income;

    }

    public void setIncome(double income) {

        this.income = income;

    }

    public String getMarried() {

        return married;

    }

    public void setMarried(String married) {

        this.married = married;

    }

    public int getChildren() {

        return children;

    }

    public void setChildren(int children) {

        this.children = children;

    }

    public String getCar() {

        return car;

    }

    public void setCar(String car) {

        this.car = car;

    }

    public String getSave\_act() {

        return save\_act;

    }

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

        this.save\_act = save\_act;

    }

    public String getCurrent\_act() {

        return current\_act;

    }

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

        this.current\_act = current\_act;

    }

    public String getMortgage() {

        return mortgage;

    }

    public void setMortgage(String mortgage) {

        this.mortgage = mortgage;

    }

    public String getPep() {

        return pep;

    }

    public void setPep(String pep) {

        this.pep = pep;

    }

    /\*

     \* Method name: readData

     \* Description: Method to read data from the CSV file

     \* Arguments: None

     \* Return type: void

     \*/

    public void readData() {//rD-S

        File file = new File("bank-Detail.csv");

        BufferedReader br;

        try {

            FileReader fReader = new FileReader(file);

            br = new BufferedReader(fReader);

            String line;

            while ((line = br.readLine()) != null) {

                // Parsing each record stored in csv file by a comma(,) into a list stored in the arraylist -> Arrays

                array.add(Arrays.asList(line.split(",")));

            }

            //Calling processData method

            processData();

        } catch (FileNotFoundException e) {

            System.out.println("Cannot read file: " + file.toString());

            System.out.println(e);

        } catch (IOException e) {

            System.out.println("Unable to read file: " + file.toString());

        }

    }//rD-E

    /\*

     \* Method name: readData

     \* Description: Method to process data from the CSV file

     \* Arguments: None

     \* Return type: void

     \*/

    public void processData() {//psD-S

        int idx = 0;

        for (List<String> rowData : array) {//FE-S

            robjs[idx] = new BankRecords();

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

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

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

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

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

            robjs[idx].setMarried(rowData.get(5));

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

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

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

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

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

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

            idx++;

        }//FE-E

        //Calling printData method

        printData();

    }//psD-E

    /\*

     \* Method name: printData

     \* Description: Method to print data from the CSV file onto console

     \* Arguments: None

     \* Return type: void

     \*/

    public void printData() {//pD-S

        /\*//Setting up welcome message

        LocalDateTime locDateTime1 = LocalDateTime.now();

        DateTimeFormatter formatter1 = DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss");

        String formatDateTime1 = locDateTime1.format(formatter1);

        System.out.println("Current date: "+ formatDateTime1);

        System.out.println("Programmed by Dhivya Udaya Kumar");

        System.out.println("Welcome to Bank of IIT Chicago Customer record generation section\n");\*/

        Scanner sc = new Scanner(System.in);

        int numRec = 0;

        do {//DW-S

            System.out.print("How many records would you like to print? : ");

            if (sc.hasNextInt()) {

                numRec = sc.nextInt();

                System.out.println();

            } else {

                sc.next();

            }

            if (numRec <= 0) {

                System.out.println("Please enter a valid integer above zero");

                System.out.println();

            }

            else {

                LocalDateTime locDateTime = LocalDateTime.now();

                DateTimeFormatter formatter = DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss.SSS");

                String formatDateTime = locDateTime.format(formatter);

                System.out.println("CSV file import started at " + formatDateTime + " milliseconds.");

                System.out.println();

                System.out.format("%-10s%-6s%-10s%-15s%-15s%-10s%-10s%-10s%-20s%-20s%-10s%-10s", "CWID", "Age", "Gender", "Locality", "Income", "Married", "Children", "Own Car", "Savings Account", "Current Account", "Mortgage", "Pep");

                System.out.println();

                int i = 0;

                for (BankRecords br : robjs) {

                    System.out.format("%-10s%-6d%-10s%-15s%-15.02f%-10s%-10d%-10s%-20s%-20s%-10s%-10s", br.getId(), br.getAge(), br.getSex(), br.getRegion(), br.getIncome(), br.getMarried(), br.getChildren(), br.getCar(), br.getSave\_act(), br.getCurrent\_act(), br.getMortgage(), br.getPep() );

                    i++;

                    System.out.println();

                    if (i == numRec) {

                        break;

                    }

                }

                LocalDateTime locDateTime2 = LocalDateTime.now();

                DateTimeFormatter formatter2 = DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss.SSS");

                String formatDateTime2 = locDateTime2.format(formatter2);

                System.out.println("\nCSV file import completed at " + formatDateTime2 + " milliseconds.");

                System.out.println("\nProgrammed by Dhivya Udaya Kumar");

                System.out.println("\nTotal records inside the bank-Detail.csv = "+array.size());

            }

        } while (numRec <= 0); //DW-E

    }//pD-E

}//class-E

1. Client.java

/\*

\* Class name: Client

\* Description: Class contains abstract methods for reading, processing and printing data from CSV file onto console

\* Programmed by: Dhivya Udaya Kumar

\* CWID: A20432502

\* Date: 3-Oct-2018

\*/

public abstract class Client {//class-S

    public abstract void readData(); //read file detail

    public abstract void processData(); //process file

    public abstract void printData(); //print file data

}//class-E

1. BankRecordsTest.java
2. /\*
3. \* Class name: BankRecordsTest
4. \* Description: Test class to test every BankRecords class methods
5. \* Programmed by: Dhivya Udaya Kumar
6. \* CWID: A20432502
7. \* Date: 3-Oct-2018
8. \*/
9. import java.io.FileNotFoundException;
10. import java.time.LocalDateTime;
11. import java.time.format.DateTimeFormatter;
12. public class BankRecordsTest {//class-S
13. /\*
14. \* Method name: main
15. \* Description: Main method of the class; Execution begins here
16. \* Arguments: String
17. \* Return type: void
18. \*/
19. public static void main(String[] args) throws FileNotFoundException {//main-S
21. //Setting up welcome message
22. LocalDateTime locDateTime1 = LocalDateTime.now();
23. DateTimeFormatter formatter1 = DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss");
24. String formatDateTime1 = locDateTime1.format(formatter1);
25. System.out.println("Current date: "+ formatDateTime1);
26. System.out.println("Programmed by Dhivya Udaya Kumar");
27. System.out.println("Welcome to Bank of IIT Chicago Customer record generation section\n");
28. //Instantiating BankRecords object
29. BankRecords bRec = new BankRecords();
31. //Calling readData method from BankRecords class
32. bRec.readData();
34. }//main-E
35. }//class-E