**PROJECT Java File Processing Application: Database Application**

**Objective** To type a simple Java program, execute ( run ) the program for some particular values, observe the output and then modify the program.

***PROJECT DESCRIPTION***

Design an application that uses sequential file processing to query a flat database file.

The initial starter code for your program is given in **Figure 1** .

First run and test the starter code, which primarily has file processing objects that declare a file and read data from the file.

After you test the original starter code and verify its functionality, you will then modify the program according to the instructions that follow.

Your completed program ( after modification ) will perform, at the minimum, each of the following tasks

• read the data from a comma separated values ( CSV ) text file

• use a split(",") function to separate the data into an ArrayList

• include a method to locate a consultant’s name in the ArrayList

• include a method to compare numerical data values in the ArrayList

• include a method to compare string data values in the ArrayList

Basically, your program is outlined in the given starter code statements shown within **Figure 1** , which follows. Review the starter code to understand the mechanisms of the interactions between reading a file and writing to a file. Perform any modifications according to this project’s instructions.

Type, compile and run the basic Java program that is shown in **Figure 1** , which follows. Then modify the program accordingly.

***Information About This Project***

In the realm of database processing, a flat file is a text file that holds a table of records.

Here is the data file that is used in this project. The data is converted to comma separated values ( CSV ) to allow easy reading into an array.

**Table: Consultants**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **LName** | **Fee** | **Specialty** |
|  |  |  |  |
| 101 | Roberts | 3500 | Media |
| 102 | Peters | 2700 | Accounting |
| 103 | Paul | 1600 | Media |
| 104 | Michael | 2300 | Web Design |
| 105 | Manfred | 3500 | Broadcasting |

***Steps to Complete This Project***

**STEP 1**  **Open NetBeans**

Open NetBeans and create a Java project with the following details.

For Project Name include **Lab14**

For the Main Class include **lab14.** **DataApplication**

In your **Code** window for this class, shown below, copy the program code shown in **Figure 1** below, in the appropriate places, except substitute your own name in place of Sammy Student.

**PROJECT Java File Processing Application: Database Application**

**Figure 1 Source Code for the Database File Processing Program**

|  |
| --- |
| **import java.io.File;**  **import java.io.FileNotFoundException;**  **import java.util.ArrayList;**  **import java.util.Scanner;**  **// Programmer: Sammy Student**  **public class DataApplication**  **{**  **public static void main(String[] args)**  **{**  **try**  **{**  **File fin = new File("data.txt");**  **Scanner scan = new Scanner(fin);**  **ArrayList<String> theData = new ArrayList<String>();**    **// read the column headings from the flat text file**  **String line = scan.nextLine();**  **while(scan.hasNextLine())**  **{**  **line = scan.nextLine();**  **String[] list = line.split(",");**  **int key = Integer.*parseInt*(list[0]);**  **String name = list[1];**  **int fee = Integer.*parseInt*(list[2]);**  **String specialty = list[3];**    **theData.add(String.*valueOf*(key));**  **theData.add(name);**  **theData.add(String.*valueOf*(fee));**  **theData.add(specialty);**  **}**  **int count = 1;**  **for (int i = 0; i < theData.size(); i++)**  **{**  **System.*out*.print(theData.get(i) + "\t\t");**  **if(count % 4 == 0 )**  **System.*out*.println(" ");**  **count++;**  **}**  **scan.close();**  **}**  **catch (FileNotFoundException e)**  **{**  **e.printStackTrace();**  **}**  **}**  **}** |

**PROJECT Java File Processing Application: Database Application**

**STEP 2 Create and Save a Text Data File**

Open a text file within your project folder. Name the text file as: **data.txt**

Copy the CSV formatted data below into the text file that you just created.

Save the data in the text file.

**[ Data File ]**

ID,LName,Fee,Specialty

101,Roberts,3500,Media

102,Peters,2700,Accounting

103,Paul,1600,Media

104,Michael,2300,Web Design

105,Manfred,3500,Broadcasting

**STEP 3 Build, Compile and Run the Program**

From the NetBeans Run menu select Run Project ( Lab14 ) to run your app.

Observe the program's output.  Notice that the data from the text file is read into an ArrayList and the array list elements are displayed in a console output statement. The data values are displayed in a tabular format.

**STEP 4 Modify Your Program**

You will now modify your database application by including a method named searchData() that receives the ArrayList elements and allows the program user to search for a name located within the flat file.

If the name of the consultant is located, a message as such is displayed by the search method otherwise a message indicating that the name is not found in the file.

The method for searching names in the flat file is shown below.  Place the method in an appropriate location in your program code.

Remember to also write a statement that calls the method.

***searchData*(theData);**

Save your program to update it and perform a trial run of your modified code.  Test your program by entering the name of a consultant that appears in the text file and run the program again using a name that is not in the file.    
 Take screen snapshots showing both a data found and data not found example.

**PROJECT Java File Processing Application: Database Application**

**Figure 2 Additional Source Code for the Database File Processing Program**

|  |
| --- |
| **public static void searchData(ArrayList<String> vals)**  **{**  **System.*out*.print("enter a name: ");**  **Scanner sc = new Scanner(System.*in*);**  **String strName = sc.nextLine().trim();**  **boolean found = false;**  **for (int i = 0; i < vals.size(); i++)**  **{**  **if(vals.get(i).equals(strName.trim()))**  **{**  **found = true;**  **break;**  **}**  **}**    **if(found == true)**  **System.*out*.println(" data found ");**  **else**  **System.*out*.println(" data not found ");**    **sc.close();**  **}** |

**STEP 5 Test the Program and Write the Data**

Now modify your program again by including a new method that will determine if any consultant charges a fee that exceeds $ 2,000 .

Save and test your program.

Finally include another method that will allow the program user to query the flat file and show a count of the consultants that specialize in providing media services.

Save and test your program.

**STEP 6 Submit Your Project**

Once you have determined that your modified program is correctly displaying the required information, complete the submission process as follows:

Open MS Word and type a heading for a new document that includes your full name, course number, lab number and date.

Within the document paste snapshots of your modified program in action. Label the snapshots of your modified run with a reasonable description.

After your snapshot, paste in your finished source code as well copied in from your NetBeans editor.

Submit your MS Word document to Blackboard when complete.