package com.lockedme; import java.io.File; import java.io.FileWriter;

import java.util.LinkedList; import java.util.Scanner;

public class LockedMe {

static String fileName;

static final String projectFilesPath = "F:\\LockedMeFiles"; static int choice = 0;

// Creating a single scanner class to be accessed in all the functions static Scanner readScanner = new Scanner(System.in);

static final String errorMessage = "Some error has occured, Please contact: [admin@Vasavi.com](mailto:admin@Vasavi.com)";

/\*\*

\* This Fucntion displays the Menu

\*/

public static void displayMenu() { System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"); System.out.println("\t\tWelcome to LockedMe.com"); System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"); System.out.println("\t\t1. Display all the files"); System.out.println("\t\t2. Add a New File");

System.out.println("\t\t3. Delete a File"); System.out.println("\t\t4. Search a File"); System.out.println("\t\t5. Exit"); System.out.println("\t\tDeveloper Details: Vasavi.N");

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

}

/\*\*

\* This method will return all the files in the directory

\*/

public static void getAllFiles() {

// using try-catch block try {

File folder = new File(projectFilesPath); File[] listOfFiles = folder.listFiles();

if (listOfFiles.length == 0) {

System.out.println("Files are not present in the directory");

} else {

for (var l : listOfFiles) {

System.out.println("File: " + l.getName());

}

}

} catch (Exception ex) {

System.out.println(errorMessage);

}

}

/\*\*

\* This method is used to create a new file into the directory

\*/

public static void createFiles() { int linesCount = 0;

// Using Try-Catch Block try {

System.out.println("Enter File Name: "); fileName = readScanner.nextLine();

FileWriter myWriter = new FileWriter(projectFilesPath + "//" + fileName); System.out.println("Enter how many Lines into the file: ");

linesCount = Integer.parseInt(readScanner.nextLine()); for (var i = 0; i <= linesCount; i++) {

System.out.println("Enter the lines you need to enter to save into

the file: ");

myWriter.write(readScanner.nextLine() + "\n");

}

System.out.println("File Successfully Created."); myWriter.close();

} catch (Exception Ex) {

System.out.println(errorMessage);

}

}

/\*\*

\* This method will delete the files to be deleted in the directory

\*/

public static void deletefiles() {

// Creating a Try Catch block try {

String fileName;

System.out.println("Enter the file name to be deleted: "); fileName = readScanner.nextLine();

File file = new File(projectFilesPath + "//" + fileName); if (file.exists()) {

file.delete();

System.out.println("File deleted successfully: " + fileName);

} else {

System.out.println("File is not present in the directory, Please create

a file in-order to "

+"see the changes in the file location");

}

} catch (Exception Ex) {

System.out.println(errorMessage);

}

}

/\*\*

\* This method will search for files in the directory

\*/

public static void searchFiles() {

// Creating a try-catch block try {

String fileName;

System.out.println("Enter the file name to be searched: "); fileName = readScanner.nextLine();

File folder = new File(projectFilesPath); File[] listOfFiles = folder.listFiles();

LinkedList<String> filenames = new LinkedList<String>(); for (var l : listOfFiles) {

filenames.add(l.getName());

}

if (filenames.contains(fileName)) { System.out.println("File is Available");

} else {

System.out.println("File not available in the directory you're

searching.");

}

} catch (Exception Ex) {

System.out.println(errorMessage);

}

}

/\*\*

* Main Method, where all the functions are being called
* @param args

\*/

public static void main(String[] args) {

// Using Do-while loop so that this application runs repeatedly do {

displayMenu();

System.out.println("Enter your choice: ");

choice = Integer.parseInt(readScanner.nextLine());

// Using Switch Case

switch (choice) { case 1:

getAllFiles(); break;

case 2:

createFiles(); break;

case 3:

deletefiles(); break;

case 4:

searchFiles(); break;

case 5:

System.exit(0); break;

default:

System.out.println("Invalid Option entered"); break;

}

} while (choice > 0);

// Closing the scanner in order to stop the leak of memory, when input is taken readScanner.close();

}

}