**Welcome.Java**

package org.lockers.com;

import java.util.Scanner;

public class Welcome {

public static void main(String[] args) {

// Welcome Page

System.out.println("Welcome to the new Lockers.com");

System.out.println("Developed By Naveenraj Venkatachalam \n");

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

System.out.println("Select the option: \n" + "1.Login \n" + "2.Signup/Register ");

int option = new Scanner(System.in).nextInt();

switch (option) {

case 1:

Boolean validation = new Login().login();

if (validation == true) {

System.out.println("Welcome to Lockers.com");

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

System.out.println("Developed By Naveenraj Venkatachalam");

System.out.println();

// Call UserInterface function

UserInterface ui = new UserInterface();

ui.userInterface();

} else {

System.out.println("You are Not a valid User, Please Retry ");

}

break;

case 2:

new Login().RegisterUser();

break;

}

}

}

**writer.java**

package org.lockers.com;

import java.io.BufferedWriter;

import java.io.FileWriter;

import java.io.IOException;

import java.io.PrintWriter;

public class writer {

public static void main(String[] args) {

String dir = System.getProperty("user.dir").concat("\\userFiles\\UserCreds");

try (FileWriter f = new FileWriter(dir, true);

BufferedWriter b = new BufferedWriter(f);

PrintWriter p = new PrintWriter(b);) {

p.println("Gaura/pass3");

} catch (IOException i) {

i.printStackTrace();

}

}

}

**FileOperations.java**

package org.lockers.com;

import java.io.File;

import java.io.FileWriter;

import java.io.IOException;

import java.util.Arrays;

import java.util.Scanner;

public class FileOperations {

UserInterface ui = new UserInterface();

Scanner s = new Scanner(System.in);

public String dir = System.getProperty("user.dir").concat("\\userFiles\\");

void listFiles() {

// dir=dir.concat("\\userFiles\\");

File directoryPath = new File(dir);

String str[] = directoryPath.list();

String temp;

System.out.println("Strings in sorted order:");

for (int j = 0; j < str.length; j++) {

for (int i = j + 1; i < str.length; i++) {

// comparing adjacent strings

if (str[i].compareTo(str[j]) < 0) {

temp = str[j];

str[j] = str[i];

str[i] = temp;

}

}

System.out.println(str[j]);

}

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

ui.userInterface();

}

void addFile() {

// Get relative path

File f = new File(dir);

f.mkdir();

System.out.println("Please Enter File name : ");

String fileName = s.next();

File stockFile = new File(dir + fileName);

try {

if (stockFile.createNewFile()) {

System.out.println(stockFile.getName() + " created succesfully");

} else {

System.out.println("File already exists.");

}

System.out.println("Do you want to insert content to your file:Y/N ");

String option = s.next();

if (option.equalsIgnoreCase("Y")) {

System.out.println("Enter contents to written in file :");

String contents = new Scanner(System.in).nextLine();

System.out.println(contents);

FileWriter writer = new FileWriter(stockFile);

writer.write(contents);

writer.close();

System.out.println("Successfully wrote to the file:" + stockFile.getName());

} else {

System.out.println("cannot be inserted");

}

} catch (IOException e) {

e.printStackTrace();

}

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

ui.userInterface();

}

void deleteFile() {

System.out.println("enter file name to delete: ");

File deleteFile = new File(dir.concat(s.next()));

if (deleteFile.delete()) {

System.out.println("File: " + deleteFile.getName() + " Deleted succesfully");

} else {

System.out.println("File not found \n");

}

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

ui.userInterface();

}

int fileSearch() {

File searchFile = new File(dir);

String[] fileNames = searchFile.list();

System.out.println(Arrays.toString(fileNames));

int min = 0, max = fileNames.length - 1;

int mid;

System.out.println("enter file name to search: ");

String key = s.next();

while (min <= max) {

mid = (min + max) / 2;

if (fileNames[mid].compareTo(key) < 0) {

min = mid + 1;

} else if (fileNames[mid].compareTo(key) > 0) {

max = mid - 1;

} else {

return 1;

}

}

return -1;

}

void businessOperations() {

// Business operations

System.out.println("select any option from below for business operation");

System.out.println(" 1) Add file");

System.out.println(" 2) Delete file");

System.out.println(" 3) Search file");

System.out.println(" 4) show previous file");

System.out.println(" 5) exit program");

// user input

// Scanner s=new Scanner(System.in);

int option = s.nextInt();

// UserInterface ui=new UserInterface();

switch (option) {

case 1:

addFile();

break;

case 2:

deleteFile();

break;

case 3:

int flag = fileSearch();

if (flag == 1) {

System.out.println("File found at directory:" + dir);

} else {

System.out.println("File Not found at directory:" + dir);

}

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

ui.userInterface();

break;

case 4:

ui.userInterface();

break;

case 5:

ui.exit();

break;

}

}

}

**Login.java**

package org.lockers.com;

import java.io.BufferedWriter;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileWriter;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.Scanner;

public class Login {

Scanner s = new Scanner(System.in);

public String dir = System.getProperty("user.dir").concat("\\userFiles\\UserCreds");

ArrayList list = new ArrayList();

boolean login() {

int count = 0;

File myObj = new File(dir);

Scanner myReader = null;

boolean flag = false;

try {

myReader = new Scanner(myObj);

while (myReader.hasNextLine()) {

String data = myReader.nextLine();

list.add(data);

}

while (count < 3) {

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

String userName = new Scanner(System.in).next();

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

String password = s.next();

String mergeCreds = userName.concat("/" + password);

for (int i = 0; i < list.size(); i++) {

if (mergeCreds.equals(list.get(i))) {

flag = true;

break;

} else

flag = false;

}

if (flag == true) {

System.out.println("Login succesfull");

break;

} else {

count++;

if (count == 3) {

new UserInterface().exit();

} else {

System.out.println("Invalid user name / password \n you will be exited from application after 3 unsuccesfull login attempts");

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

}

}

}

// myReader.close();

} catch (FileNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return flag;

}

void RegisterUser() {

Scanner s = new Scanner(System.in);

String dir = System.getProperty("user.dir").concat("\\userFiles\\UserCreds");

System.out.println("Enter UserName to Register:");

String userName = s.next();

System.out.println("Enter Password to Register:");

String pass = s.next();

s.close();

String mergeCreds = userName.concat("/" + pass);

try (FileWriter f = new FileWriter(dir, true);

BufferedWriter b = new BufferedWriter(f);

PrintWriter p = new PrintWriter(b);) {

p.println(mergeCreds);

System.out.println("Succesfully registered to locker.com\n");

System.out.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");

} catch (IOException i) {

i.printStackTrace();

}

}

}

**UserInterface.java**

package org.lockers.com;

import java.util.Scanner;

public class UserInterface {

void exit()

{

System.*out*.println("Thanks, You have sucessfully Exited from your source code programme !");

}

//Prompt user interface options.

void userInterface()

{

System.*out*.println("Select any option from below ");

System.*out*.println("1. Retrieving the file names ");

System.*out*.println("2. Menu for File operation ");

System.*out*.println("3. Close application ");

//Read Input from user

Scanner input=new Scanner(System.*in*);

int option= input.nextInt();

switch(option)

{

case 1: new FileOperations().listFiles(); break;

case 2:new FileOperations().businessOperations(); break;

case 3: exit(); break;

}

input.close();

}

}