package Virtual\_Key\_Repositories;

import java.util.Arrays;

import java.io.File;

import java.io.IOException;

import java.util.Scanner;

class DisplayFile {

// Display file(ascending order)....

public void displayList() {

String path = "C:\\Users\\parg\\OneDrive\\Desktop\\FSD\_Phase1\\Virtual\_Key\_Repo";

File fl = new File(path);

File[] files = fl.listFiles();

// sorted files in ascending order...

Arrays.sort(files);

// show...

System.out.println(" > Your files & directory in " + fl.getName() + "'s are : \n");

for (File f : files) {

System.out.println(" - " + f.getName());

}

System.out.println();

}

}

class AddFile {

String path ="C:\\Users\\parg\\OneDrive\\Desktop\\FSD\_Phase1\\Virtual\_Key\_Repo";

Scanner sc = new Scanner(System.in);

public void createFile() {

System.out.print("Enter the file name to create : ");

String fileName = sc.next();

String filePath = path + fileName; // complete path with file name...

File f = new File(filePath);

try {

if (f.createNewFile()) { // create new file....

System.out.println("file is created successfully...\n");

} else {

System.out.println("file not created! already Exist....");

}

} catch (IOException e) {

e.printStackTrace(); // throw exception...

}

}

}

class DeleteFile {

String path = "C:\\Users\\parg\\OneDrive\\Desktop\\FSD\_Phase1\\Virtual\_Key\_Repo";

Scanner sc = new Scanner(System.in);

public void deleteFile() {

DisplayFile df = new DisplayFile();

df.displayList();

System.out.print("Enter the file name for deletetion... : ");

String fileName = sc.next();

String filePath = path + fileName; // complete path with file name...

File f = new File(filePath);

if (f.delete()) { // delete the file....

System.out.println("file deleted successfully...");

} else {

System.out.println("file not found....");

}

}

}

class SearchFile {

Scanner sc = new Scanner(System.in);

public void searchFile() {

String path = "C:\\Users\\parg\\OneDrive\\Desktop\\FSD\_Phase1\\Virtual\_Key\_Repo";

DisplayFile df = new DisplayFile();

df.displayList(); // call displayList() from the DisplayFile Class..

System.out.print("Enter the file name for search.. : ");

String fileName = sc.next();

String filePath = path + fileName;

File f = new File(filePath);

File fl = new File(path);

File[] files = fl.listFiles(); // keep all files and dir. to files array..

int flag = 0;

for (File fp : files) {

if (fp.getName().equals(fp.getName())) {

flag = 1;

break;

} else {

flag = 0;

}

}

if (flag == 1) {

System.out.println("your file \'" + f.getName() + "\' found...");

} else {

System.out.println(" - File not found... ");

}

}

}

public class Main {

public static void main(String[] args) throws IOException {

// all classes object creation...

SearchFile sf = new SearchFile();

DeleteFile df = new DeleteFile();

AddFile af = new AddFile();

DisplayFile dsf = new DisplayFile();

Scanner sc = new Scanner(System.in);

// infinite while loop used to iterate until user want to run...

label: while (true) {

System.out.println(

"\t----------------------------\t\tVirtual Key For Your Repositories\t------------------------");

System.out.println("Developer : Pargati");

System.out.print(

"\n> Select '1' to see your file...\n> Select '2' to dooperation on file...\n> Select '3' to Exit...\n\n\*\* Input :");

int choice = sc.nextInt();

switch (choice) {

case 1:

dsf.displayList(); // display the file....

break;

case 2:

while (true) {

System.out.println("> ----------------- OperationsPerformingOptions -------------------");

System.out.print(

" - 1. Add File...\n - 2. Search File...\n - 3. Delete File...\n - 4. Back to Menu...\nChoose the Options : ");

int opt = sc.nextInt();

System.out.println();

switch (opt) {

case 1:

af.createFile();// Adding file in File .....

break;

case 2:

sf.searchFile();// Searching file From FIle.....

break;

case 3:

df.deleteFile();// Deleting file From.....

break;

case 4:

continue label; // used to go to backmenu...

default:

System.out.println("Incorrect options\nPlease try again.");

break;

}

}

case 3:

sc.close();

System.exit(0);

default:

System.out.println("Incorrect options\nPlease try again.");

break;

}

}

}

}