**SOURCE CODE for Phase1 Project**

**MainApp.java**

**package** FileEntryConsole;

**import** java.io.IOException;

// Main class of the Application

**public** **class** MainApp {

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

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

System.***out***.println("Welcome To Lockers Pvt File Viewer and Manager Developed by Ajunumasa");

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

FileAppManager fm = **new** FileAppManager();

fm.filemanager();

}

}

**FileAppManager.java**

package FileEntryConsole;

import FileOperations.displayFile;

import FileOperations.searchFile;

import FileOperations.deleteFile;

import FileOperations.addFile;

import java.io.IOException;

import java.util.Scanner;

public class FileAppManager {

public void filemanager() throws IOException{

try

{

boolean exit = false;

Scanner scChoice = new Scanner(System.in);

int optionChoosen = 0;

do {

System.out.println("");

System.out.println("Choose from 1 to 5 and press Enter to perform the below operations");

System.out.println("");

System.out.println("1-Display Files in Sorted order");

System.out.println("2-Add a New File");

System.out.println("3-Delete File");

System.out.println("4-Search For a File");

System.out.println("5-Exit");

optionChoosen = scChoice.nextInt();

switch (optionChoosen) {

case 1:

displayFile obj = new displayFile();

obj.displayFileFunc("root");

break;

case 2:

System.out.println("Enter the File Name to Add");

Scanner scAdd = new Scanner(System.in);

String FileNameToAdd = scAdd.nextLine();

addFile obj4=new addFile();

obj4.AddFileFunc("root",FileNameToAdd.trim());

break;

case 3:

System.out.println("Enter the File Name to Delete");

Scanner scDel = new Scanner(System.in);

String FileToDelete = scDel.nextLine();

deleteFile obj3 = new deleteFile();

obj3.deleteFileFunc("root", FileToDelete.trim());

break;

case 4:

System.out.println("Enter the File Name to Search");

Scanner scSearch = new Scanner(System.in);

String FileToSearch = scSearch.nextLine();

searchFile obj2 = new searchFile();

obj2.searchFile("root", FileToSearch.trim());

break;

case 5:

exit = true;

System.out.println("Thank you for using Lockers Pvt.com SEE YOU SOON!!");

break;

default:

System.out.println("Incorrect option : Choose the option from 1 to 5 to perform the below operations");

exit = true;

break;

}

}

while (!exit);

}

catch (Exception e){

System.out.println(" "+"PLEASE CHECK THE OPTION ENTERED");

}

}

}

**DisplayFile.Java**

package FileOperations;

import java.io.File;

import java.io.IOException;

import java.nio.file.Files;

import java.nio.file.Paths;

public class displayFile implements OperationsDisplay {

// This method shows files in the directory in sorted order

@Override

public void displayFileFunc(String FolderPath) throws IOException {

Files.list(Paths.get(FolderPath)).filter(Files::isRegularFile).sorted().forEach(System.out::println);

}

// This method returns the countOfFiles in the directory

@Override

public int countFilesFunc(String FolderPath) {

File FileCount=new File(FolderPath);

String files[]=FileCount.list();

return files.length;

}

//This method checks whether the directory is empty or not

@Override

public boolean isEmpty(String FolderPath) {

File FileCount=new File(FolderPath);

String files[]=FileCount.list();

if (files.length>0){

return false;

}

else {

return true;

}

}

}

interface OperationsDisplay {

public void displayFileFunc(String FolderPath) throws IOException;

public int countFilesFunc(String FolderPath);

public boolean isEmpty(String FolderPath);

}

**AddFile.java**

package FileOperations;

import java.io.IOException;

import java.nio.file.Files;

import java.nio.file.Paths;

// This method adds a File to the directory.It also checks whether a file already exists and if a file exists then it rejects

public class addFile implements OperationsAdd{

@Override

public void AddFileFunc(String Foldername, String FileName) throws IOException {

displayFile obj = new displayFile();

int count = obj.countFilesFunc(Foldername);

if (count >= 0) {

if(!Files.exists(Paths.get(Foldername,FileName))){

Files.createFile(Paths.get(Foldername,FileName));

System.out.println("File created successfully");

}

else{

System.out.println("File Already Exists in the Directory");

}

}

}

}

interface OperationsAdd {

public void AddFileFunc(String Foldername,String FileName) throws IOException;

}

**DeleteFile.java**

package FileOperations;

import java.io.IOException;

import java.nio.file.Files;

import java.nio.file.Path;

import java.nio.file.Paths;

public class deleteFile implements OperationsDelete {

//This method deletes file if it exists in the directory else it will display file doesn't exist.

@Override

public void deleteFileFunc(String folderpath, String filename) throws IOException {

displayFile obj = new displayFile();

int count = obj.countFilesFunc(folderpath);

if (count > 0)

{

Path path = Paths.get(folderpath, filename);

if (Files.exists(path))

{

Files.deleteIfExists(Paths.get(folderpath, filename));

System.out.println("Deleted Successfully");

}

else {

System.out.println("File Doesn't Exist");

}

}

else

{

System.out.println("Directory is Empty");

}

}

}

interface OperationsDelete {

public void deleteFileFunc(String folderpath,String filename) throws IOException;

}

**SearchFile.java**

package FileOperations;

import java.io.File;

import java.io.IOException;

public class searchFile implements OperationsSearch{

// This method searches for a file in the directory irrespective of case -sensitivity

@Override

public String searchFile(String FolderPath,String FileName) throws IOException {

displayFile obj =new displayFile();

int count=obj.countFilesFunc(FolderPath);

String DirectoryName=FolderPath+"\\";

String fileFound=null;

if(count>0) {

File file =new File(DirectoryName+FileName);

if(file.exists()){

fileFound=DirectoryName+FileName;

System.out.println("File Exists"+" "+DirectoryName+FileName);

}

else{

fileFound=null;

System.out.println("File Not Found");

}

}

else{

System.out.println("Directory is empty");

}

return fileFound;

}

}

interface OperationsSearch {

public String searchFile(String FolderPath,String FileName) throws IOException;

}

\*\*\*\* end of Source code \*\*\*\*