Virtual Key for Your Repositories
Lockers pvt. Ltd
Prototype of the application for lockedme.com

Locked Me.Com Project Developed by Kadham Santosh Details are –

Sprints planning and Task completion-

Project work dived into 2 Sprint 10 days, Sprint work details are –

1st – Sprint Work Details -

A) Initialize the Git Repository to track changes as development progresses.
B) Create Java class for show the files from folder in ascending order.
C) Create Java class for Option to add a user specified file to the application.
D) Create Java class for Option to search a user specified file from the application with case sensitive.
E) Create Java class for Option to read a user specified file from the application with case sensitive.
F) Push the Java class in git repository.

2nd – Sprint Work Details –

A) Create Java class for Option to delete user specified file from the application with case sensitive.
B) Create Java class for exit from application.
C) Create Main Java class as a welcome class to Navigation option to close the current execution context and return to the main context.
D) Welcome Class is main class to execute all the classes.
E) Test Again All the Java class.
F) Migrate all the Java class to welcome class and execute the application.
G) Test application make the required changes.
H) Push the Java class in git repository.

Core concepts used in project-

Collection's framework, File Handling, Sorting, Flow Control, Recursion, Exception Handling, Loop, Algorithm, sorting etc.

Welcome class is main class of application to execute the application. In this class created all classes used as object, while loop, switch loop etc. to execute all function of application.

Java Classes to use in project –

1. **WelcomeLockedme Class** - Use to welcome the user, Main menu and operational menu.
2. **ShowFile Class –** Use to Show file name, file name in ascending order.
3. **AddNewFile class –** Use to create new file, input taken from user for file name and some write up in file.
4. **ReadFile class –** To read existing file with file name is case sensitivity.
5. **SearchFile class -** To search existing file with file name is case sensitivity.
6. **DeleteFile class –** To delete existing file with file name is case sensitivity.
7. **Exit class** – To exit the application.
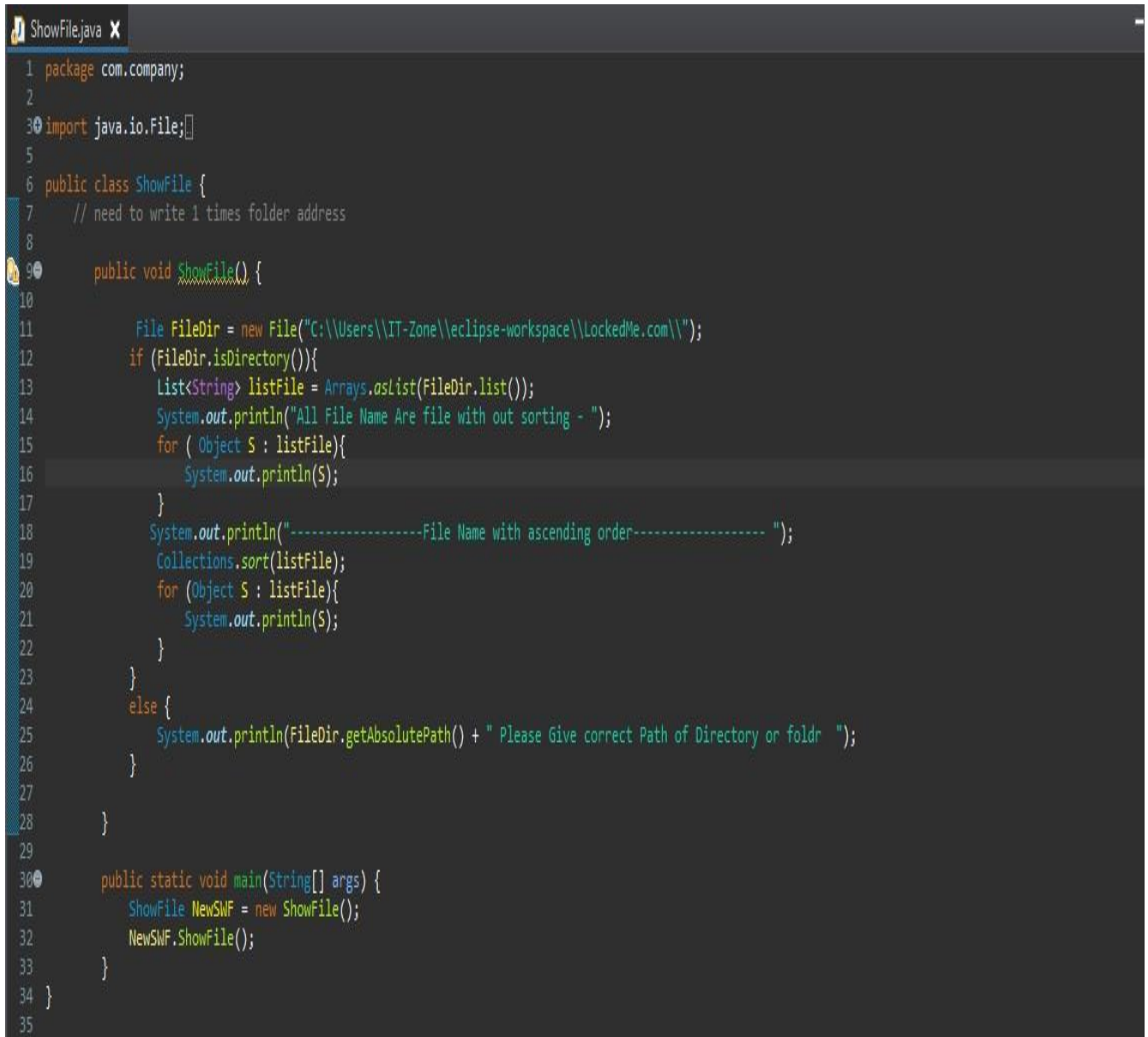
Virtual Key for Your Repositories Lockers pvt. Ltd

Details Concept for Each class with code and Result –

1. **ShowFile Class:-**

In this class used File handling, sorting, Exception handling, Array, If and for loop. Class used to show file without shorting and thane file on ascending order.

Code Image Is –

```java
package com.company;

import java.io.File;

public class ShowFile {
    // need to write 1 times folder address

    public void ShowFile() {

        File FileDir = new File("C:\\Users\\IT-Zone\\eclipse-workspace\\LockedMe.com\\");
        if (FileDir.isDirectory()){
            List<String> listFile = Arrays.asList(FileDir.list());
            System.out.println("All File Name Are file with out sorting - ");
            for ( Object S : listFile){
                System.out.println(S);
            }
            System.out.println("-------------------File Name with ascending order----------------- ");
            Collections.sort(listFile);
            for (Object S : listFile){
                System.out.println(S);
            }
        }
        else {
            System.out.println(FileDir.getAbsolutePath() + " Please Give correct Path of Directory or foldr  ");
        }

    }

    public static void main(String[] args) {
        ShowFile NewSWF = new ShowFile();
        NewSWF.ShowFile();
    }
}
```

Output Result for Showclass.java –

```
Console ✕
<terminated> ShowFile [Java Application] C:\Users\IT-Zone\.p2\pool\plugins\org.eclipse.just
All File Name Are file with out sorting -
-multiinst
.classpath
.git
.project
bin
dass.txt
hi.txt
san.txt
src
--------------------File Name with ascending order--------------------
-multiinst
.classpath
.git
.project
bin
dass.txt
hi.txt
san.txt
src
```

2. AddNewFile Class –

In this class used File Handling, Scanner to get input from user, Exception Handling, Array and loop. Class Used to Add new file name given by user and Write some data in file by user.

Code Imege –
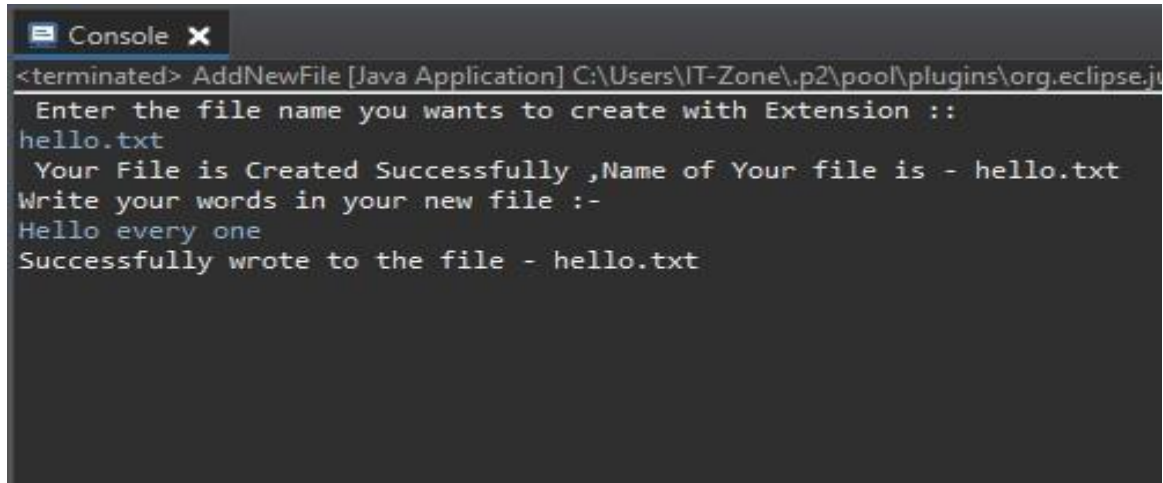
```java
ShowFile.java      AddNewFile.java ✕
1  package com.company;
2
3  import java.io.File;
7
8  public class AddNewFile {
9
10
11     public void AddNewFile() throws IOException {
12         //2 times need to give folder address
13
14         File FileDir = new File("C:\\Users\\IT-Zone\\eclipse-workspace\\LockedMe.com\\" );
15         if(FileDir.isDirectory()) {
16             Scanner Filename = new Scanner(System.in);
17             System.out.println(" Enter the file name you wants to create with Extension :: ");
18             String Newfile1 = Filename.nextLine();
19             File Newfile = new File("C:\\Users\\IT-Zone\\eclipse-workspace\\LockedMe.com\\" + Newfile1);
20             Newfile.createNewFile();
21             System.out.println(" Your File is Created Successfully ,Name of Your file is - " + Newfile1);
22             FileWriter writfile = new FileWriter(Newfile);
23             System.out.println("Write your words in your new file :- ");
24             String Write = Filename.nextLine();
25             writfile.write(Write);
26             writfile.close();
27             System.out.println("Successfully wrote to the file - "+ Newfile1);
28         }
29         else {
30             System.out.println(FileDir.getAbsolutePath() + " Please Give correct Path of Directory or folder ");
31         }
32     }
33     public static void main(String[] args) throws IOException {
34         AddNewFile addfile = new AddNewFile();
35         addfile.AddNewFile();
36     }
37  }
38
39
```

Out Put Result for AddNewFileClass –

3. ReadFile Class –

In this class concept use of File handling, Scanner for user input, Exception handling, Array, collection, If and for loop, Case sensitive concept. Class used for Read existing file with file name is case sensitive.

Code Image Is –

```java
package com.company;

import java.io.File;
import java.io.FileNotFoundException;
import java.util.Arrays;
import java.util.List;
import java.util.Scanner;

public class Readfile {
    public void Readfile() throws FileNotFoundException {
        // Need to give 2 times folder address in code
        File FileDir = new File("C:\\Users\\IT-Zone\\eclipse-workspace\\LockedMe.com\\");
        List<String> listFile = Arrays.asList(FileDir.list());
        File[] filelist = FileDir.listFiles();
        if (FileDir.isDirectory()) {
            Scanner FileName = new Scanner(System.in);
            System.out.println("Enter the file name wants to Read (File name is Case sensitive), with Extension( Like- .txt,.doc etc) :- ");
            String RF = FileName.nextLine();
            File Redfile = new File("C:\\Users\\IT-Zone\\eclipse-workspace\\LockedMe.com\\" + RF);
            boolean a = false;
            if (filelist == null) throw new AssertionError("File Not found in Folder");
            for (File I : filelist) {
                String k = I.getName();
                if (k.equals(RF)) {
                    a = true;
                    break;
                }
            }
            if (!a) {
                System.out.println("File not Present in folder");
            }
            else {
                Scanner sc = new Scanner(Redfile);
                while (sc.hasNext()) {
                    System.out.println(sc.nextLine());
                }
            }

        }
        else {
            System.out.println(FileDir.getAbsolutePath() + " Please Give correct Path of Directory or folder ");
        }
```

```
40          else {
41              System.out.println(FileDir.getAbsolutePath() + " Please Give correct Path of Directory or folder ");
42          }
43      }
44
45●     public static void main(String[] args) throws FileNotFoundException {
46          Readfile read = new Readfile();
47          read.Readfile();
48
49      }
50  }
51
```

Out Put Result for ReadFile Class –

A) If File Name is correct –

Console ✕

<terminated> Readfile [Java Application] C:\Users\IT-Zone\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.1.v20
```
Enter the file name wants to Read (File name is Case sensitive), with Extension( Like- .txt,.doc etc) :-
hello.txt
Hello every one
```

B) If File Name is wrong –

Console ✕

<terminated> Readfile [Java Application] C:\Users\IT-Zone\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.1.v20
```
Enter the file name wants to Read (File name is Case sensitive), with Extension( Like- .txt,.doc etc) :-
qwer.txt
File not Present in folder
```

4) SearchFile Class –

In this class concept use of File handling, Scanner for user input, Exception handling, Array, collection, If and for loop, Case sensitive concept. Class used for Search existing file with file name is case sensitive .

Virtual Key for Your Repositories
Lockers pvt. Ltd

Code Image Is –

```java
package com.company;

import java.io.File;
import java.util.Arrays;
import java.util.List;
import java.util.Scanner;

public class SearchFile {
    public void SearchFile(){
        // need to write 2 times folder address

        File FileDir = new File("C:\\Users\\IT-Zone\\eclipse-workspace\\LockedMe.com\\");
        List<String> listFile = Arrays.asList(FileDir.list());
        File[] filelist = FileDir.listFiles();
            if (FileDir.isDirectory()) {
                Scanner SearchFile = new Scanner(System.in);
                System.out.println("Enter the file name(File name is case sensitive) wants to Search With Extension( Like- .txt,.doc etc) :- ");
                String serfil = SearchFile.nextLine();
                File Serfile = new File("C:\\Users\\IT-Zone\\eclipse-workspace\\LockedMe.com\\" + serfil);
                boolean B = false;
                if (filelist == null) throw new AssertionError("File Not found in Folder");
                for (File i : filelist) {
                    String SF = i.getName();
                    if (SF.equals(serfil)) {
                        B = true;
                        System.out.println("Successfully completed search of file  = " + serfil+"\n File address is ="+i.getAbsolutePath());
                        break;
                    }
                }
                if (!B){
                System.out.println(" File Not found ");
                }
            }
            else {
            System.out.println(FileDir.getAbsolutePath() + " Please Give correct Path of Directory or folder ");
            }
        }
        public static void main(String[] args) {
            SearchFile FSE = new SearchFile();
            FSE.SearchFile();
    }
}
```

Out Put Result for SearchFile Class –

A) File Present in Folder –

```
Enter the file name(File name is case sensitive) wants to Search With Extension( Like- .txt,.doc etc) :-
san.txt
Successfully completed search of file  = san.txt
 File address is =C:\Users\IT-Zone\eclipse-workspace\LockedMe.com\san.txt
```

B) If File Not Present in Folder or Spell Mistake –

```
Enter the file name(File name is case sensitive) wants to Search With Extension( Like- .txt,.doc etc) :-
sant.txt
 File Not found
```
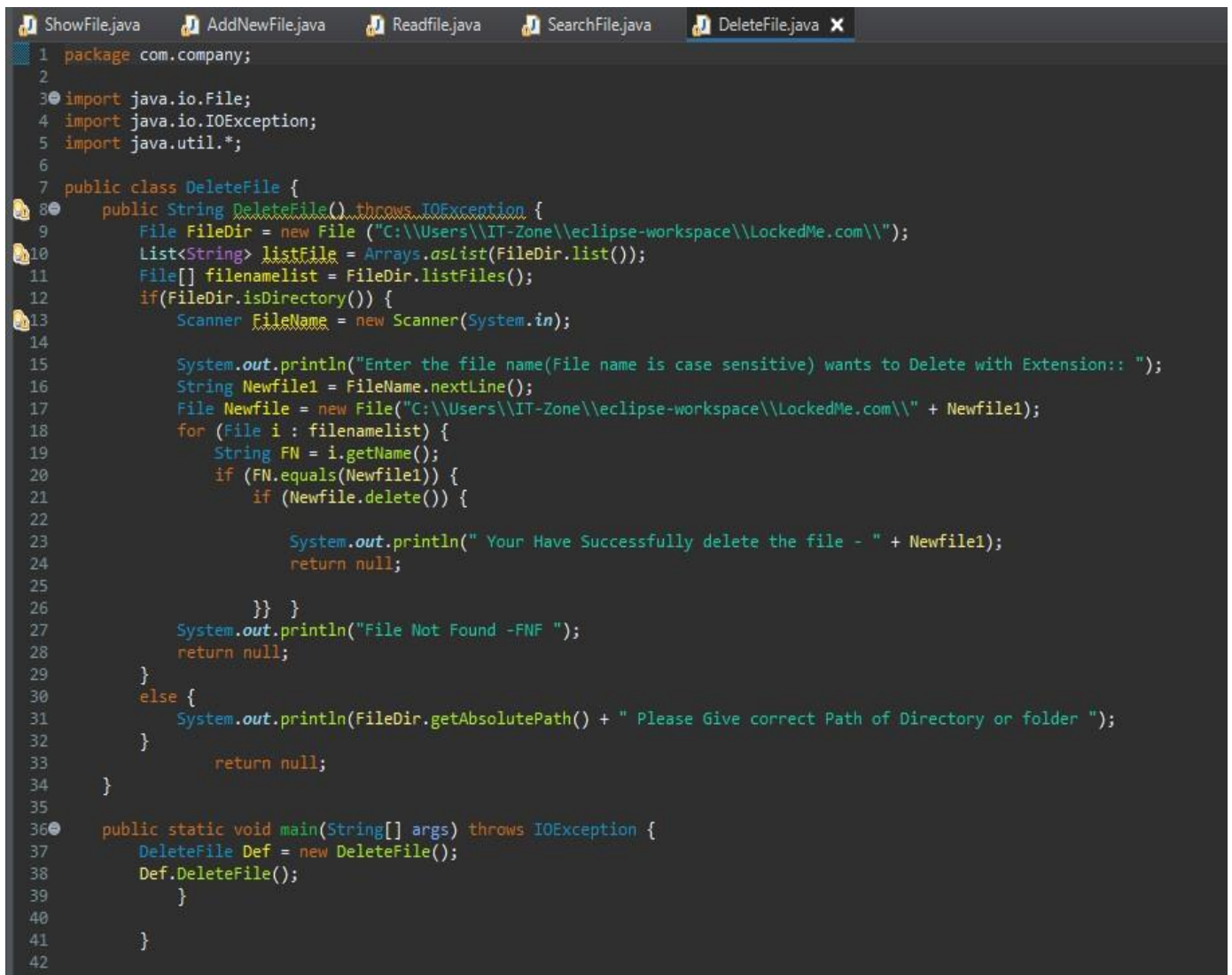
5) DeleteFile Class –

In this class concept use of File handling, Scanner for user input, Exception handling, Array, collection, If and for loop, Case sensitive concept. Class used for Delete existing file with file name is case sensitive.
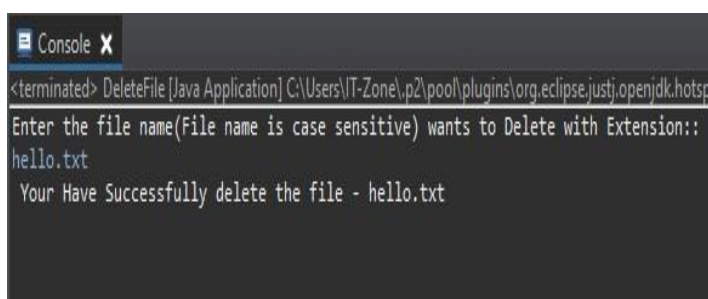
Code Image Is –

```java
package com.company;

import java.io.File;
import java.io.IOException;
import java.util.*;

public class DeleteFile {
    public String DeleteFile() throws IOException {
        File FileDir = new File ("C:\\Users\\IT-Zone\\eclipse-workspace\\LockedMe.com\\");
        List<String> listFile = Arrays.asList(FileDir.list());
        File[] filenamelist = FileDir.listFiles();
        if(FileDir.isDirectory()) {
            Scanner FileName = new Scanner(System.in);

            System.out.println("Enter the file name(File name is case sensitive) wants to Delete with Extension:: ");
            String Newfile1 = FileName.nextLine();
            File Newfile = new File("C:\\Users\\IT-Zone\\eclipse-workspace\\LockedMe.com\\" + Newfile1);
            for (File i : filenamelist) {
                String FN = i.getName();
                if (FN.equals(Newfile1)) {
                    if (Newfile.delete()) {

                        System.out.println(" Your Have Successfully delete the file - " + Newfile1);
                        return null;

                }} }
            System.out.println("File Not Found -FNF ");
            return null;
        }
        else {
            System.out.println(FileDir.getAbsolutePath() + " Please Give correct Path of Directory or folder ");
        }
            return null;
    }

    public static void main(String[] args) throws IOException {
        DeleteFile Def = new DeleteFile();
        Def.DeleteFile();
        }

    }
```
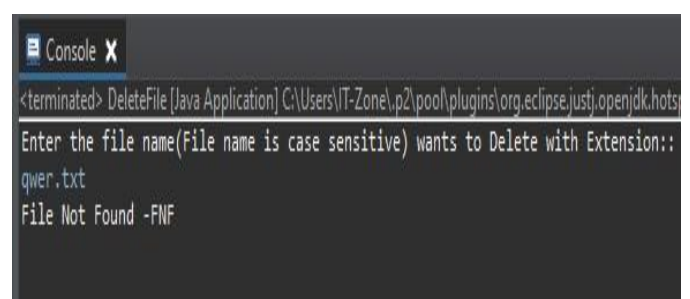
Out Put Result for DeleteFile Class –

A) File name is correct   B) File Name is Wrong

Console X
<terminated> DeleteFile [Java Application] C:\Users\IT-Zone\.p2\pool\plugins\org.eclipse.justj.openjdk.hotsp
Enter the file name(File name is case sensitive) wants to Delete with Extension::
hello.txt
Your Have Successfully delete the file - hello.txt

Console X
<terminated> DeleteFile [Java Application] C:\Users\IT-Zone\.p2\pool\plugins\org.eclipse.justj.openjdk.hots
Enter the file name(File name is case sensitive) wants to Delete with Extension::
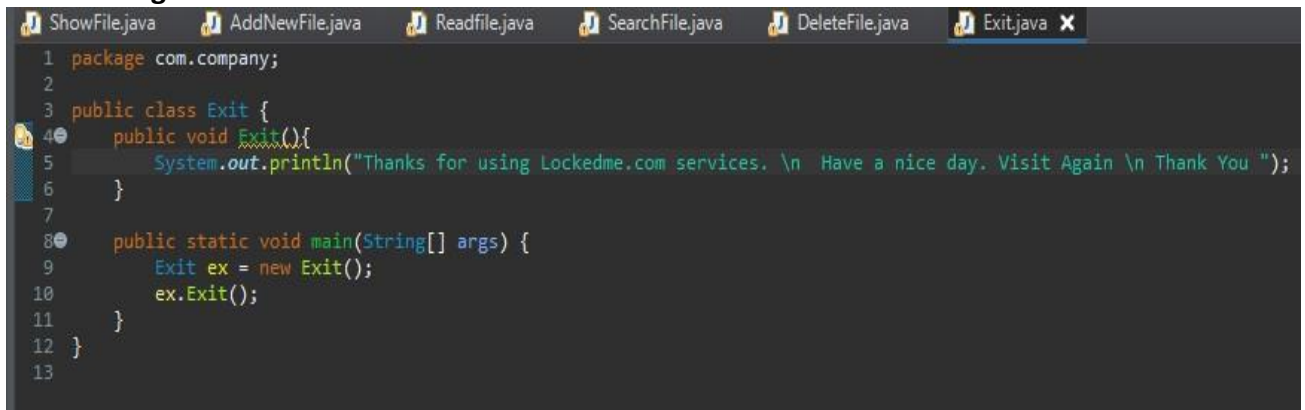qwer.txt
File Not Found -FNF

6) Exit Class –
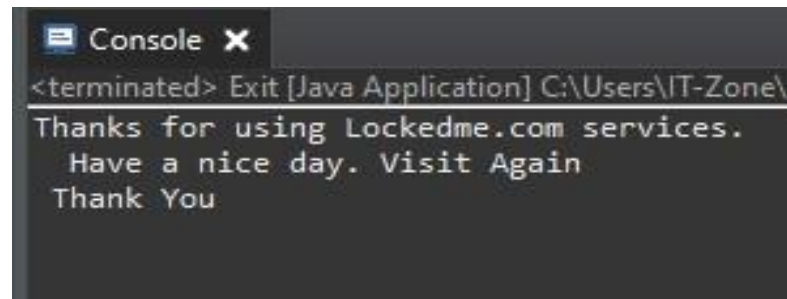
Very simple concept.

Code Image is –

Out Put Result for Exit Class –

7) WelcomeLockedme Class –

This is very Important Class Were application is executed, in this class all above 6 class used as an object, While, If, Switch, Exception and Scanner. Concept of class are -

A) First this class shows welcome to user, developer name.
B) Show Main menu to Enter the File Operation.
C) In File operation menu show all the operation of application i.e.-

```
1)  Enter - s - To Show all file.
2) Enter - c - To create a new file
3) Enter - d - To Delete a File
4) Enter - r - To Read File
5) Enter - f - To Search (Find) Existing File
6) Enter - e - To Exit from Application
```

D) Select the Option from operation menu and it will perform mention operation.

Virtual Key for Your Repositories
Lockers pvt. Ltd

E) After Performing operation, it will return to main menu if you want continue or exit.

F) If wants continue same operation menu will open as motion on C).

G) Don't want to continue type e and it will exit the application.

H) Below the code and output result in Image.

Code for WelcomeLockedme Class

```java
package com.company;

import java.io.IOException;
import java.util.Scanner;

public class WelcomeLockedme {
    public static void main(String[] args) throws IOException {
        System.out.println("Welcome from Lockers Pvt. Ltd.\n Application for File Operations");
        System.out.println("Develop by :- KADHAM SANTOSH ");
        Scanner WC = new Scanner(System.in);

        while (true) {
            System.out.println("");
            System.out.println("Type y to Continue the File operation or e to exit:-");
            String ask = WC.nextLine();
            if (ask.equals("y")) {
                System.out.println(" Choose Below option as per your Requirement\n Option Menu Are:-");
                System.out.println("Enter - s - To Show all file. \nEnter - c - To create a New file \nEnter - d - To Delete a File " +
                        "\nEnter - r - To Read File\nEnter - f - To Search(Find) Existing File\nEnter - e - To Exit from Application  ");
                System.out.println("Please select correct option to perform the application :- ");
            } else if (ask.equals("e")) {
                Exit ex = new Exit();
                ex.Exit();
                break;
            } else {
                System.out.println(" Invalid option please choose correct option and restart the application ");
                return ;
            }
            String as = WC.nextLine();
            if (as.equals("e")){
                Exit EX = new Exit();
                EX.Exit();
                break;
            }
```

```java
            String as = WC.nextLine();
            if (as.equals("e")){
                Exit EX = new Exit();
                EX.Exit();
                break;
            }
            switch (as) {
                case "s":
                    ShowFile sow = new ShowFile();
                    sow.ShowFile();
                    break;
                case "c":
                    AddNewFile crt = new AddNewFile();
                    crt.AddNewFile();
                    break;
                case "d":
                    DeleteFile de = new DeleteFile();
                    de.DeleteFile();
                    break;
                case "r":
                    Readfile re = new Readfile();
                    re.Readfile();
                    break;
                case "f":
                    SearchFile sf = new SearchFile();
                    sf.SearchFile();
                    break;

                default:
                    System.out.println(" Invalid option please choose correct option form Option Menu ");
            }
        }
    }
}
```

Virtual Key for Your Repositories
Lockers pvt. Ltd

Out Put Result for WelcomeLockedme Class -

A) Welcome to user , Main menu and Operation menu –

B) Type s for Show all file –

C) Type c to create new file   **D) Type r to read file**

E) Type f to search file –

```
Console ✕
WelcomeLockedme [Java Application] C:\Users\IT-Zone\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.1.v20210
 Option Menu Are:-
Enter - s - To Show all file.
Enter - c - To create a New file
Enter - d - To Delete a File
Enter - r - To Read File
Enter - f - To Search(Find) Existing File
Enter - e - To Exit from Application
Please select correct option to perform the application :-
f
Enter the file name(File name is case sensitive) wants to Search With Extension( Like- .txt,.doc etc) :-
hello.txt
Successfully completed search of file  = hello.txt
 File address is =C:\Users\IT-Zone\eclipse-workspace\LockedMe.com\hello.txt

Type y to Continue the File operation or e to exit:-
```

F) Type d to delete file –

```
Console ✕
WelcomeLockedme [Java Application] C:\Users\IT-Zone\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.fu
 Choose Below option as per your Requirement
 Option Menu Are:-
Enter - s - To Show all file.
Enter - c - To create a New file
Enter - d - To Delete a File
Enter - r - To Read File
Enter - f - To Search(Find) Existing File
Enter - e - To Exit from Application
Please select correct option to perform the application :-
d
Enter the file name(File name is case sensitive) wants to Delete with Extension::
hello.txt
 Your Have Successfully delete the file - hello.txt

Type y to Continue the File operation or e to exit:-
```

G) Type e to exit the application –

```
Console ✕
<terminated> WelcomeLockedme [Java Application] C:\Users\IT-Zone\.p2\pool\
 Type y to Continue the File operation or e to exit:-
 y
  Choose Below option as per your Requirement
  Option Menu Are:-
Enter - s - To Show all file.
Enter - c - To create a New file
Enter - d - To Delete a File
Enter - r - To Read File
Enter - f - To Search(Find) Existing File
Enter - e - To Exit from Application
Please select correct option to perform the application :-
e
Thanks for using Lockedme.com services.
  Have a nice day. Visit Again
 Thank You
```

Virtual Key for Your Repositories
Lockers pvt. Ltd

Note :- After every class push the code to git hub git hub link is –

https://github.com/santosh4246/Phase1-Project.git
or

https://github.com/santosh4246/lockedme.com.git

Unique Selling Points of the Application

1. The application is designed to keep on running and taking user inputs even after exceptions occur. To terminate the application, appropriate option needs to be selected.

2. The application can take any file/folder name as input. Even if the user wants to create nested folder structure, user can specify the relative path, and the application takes care of creating the required folder structure.

3. User is also provided the option to write content if they want into the newly created file.

4. The user is able to seamlessly switch between options or return to previous menu even after any required operation like adding, searching or deleting of files is performed.

5. The application is designed with modularity in mind. Even if one wants to update the path, they can change it through the source code. Application has been developed keeping in mind that there should be very less "hardcoding" of data.

If any issue please fill free to contact me my details are-

<div align="center">

Thanks, Develop by

KADHAM SANTOSH

Email id – ksantosh0305@gmail.com

Mobile No- 9618204246

</div>