



LOCKEDME.COM

Submitted by Manvendra Pratap Singh



BATCH AND PHASE
JFS FEB 14 COHORT – PHASE 1
Simplilearn

Contents

Technology	2
Java Concept Used	2
Project Folder Structure.....	2
Source Code	2
Project Screen Shot.....	7
Git Repository URL	10

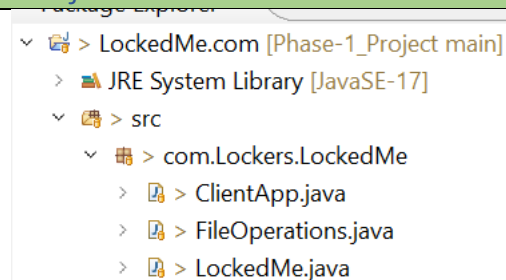
Technology

- Core Java and Data Structure

Java Concept Used

- File handling
- Exception Handling
- Java Object Oriented Programming
- Java Collection
- Access Modifiers
- Control Structure
- Naming Standard
- Re-usability

Project Folder Structure



```
Package Explorer
└─ LockedMe.com [Phase-1_Project main]
   └─ JRE System Library [JavaSE-17]
      └─ src
         └─ com.Lockers.LockedMe
            ├── ClientApp.java
            ├── FileOperations.java
            └── LockedMe.java
```

Source Code

- ClientApp.java

```
package com.Lockers.LockedMe;

import java.util.Scanner;
/**
 * Class Name: ClientApp
 * This contain main of application
 * Inside class
 * Method Name: main
 * Inside method:
 * 1. Display welcome screen and main menu
 * 2. Take input from user form menu options
```

```

* 3. Perform action according to user input
* a.retrieve all file
* b.file operations
* c.exit
* @author manvendra
*
*/
public class ClientApp {

    public static void main(String[] args) {

        try {
            // Variable declaration
            int choice;
            try (Scanner scan = new Scanner(System.in)) {
                do {
                    // Displaying main welcome screen and main menu
                    LockedMe.displayMainMenu();

                    // Asking for user choice
                    System.out.println("Please enter your choice");

                    // Storing user input
                    choice = Integer.parseInt(scan.nextLine());

                    // Checking user input and perform action accordingly
                    switch(choice) {
                        case 1 -> LockedMe.retrieveAllFiles();
                        case 2 -> LockedMe.fileOperations(scan);
                        case 3 -> LockedMe.exit();
                        default -> System.out.println("Invalid input");
                    }
                }while(true);
            }
        }

        catch (Exception e) {
            System.out.println(e.getMessage());
        }

    }

}

```

➤ FileOperations.java

```

package com.Lockers.LockedMe;

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

/**
 * Class Name: FileOperations
 * This class contain all methods related to file operation
 * Inside Class:
 * 1. method addFile() -> to add file
 * 2. method deleteFile() -> to delete file
 * 3. method searchFile() -> to search file
 * @author Manvendra
 */

```

```

*/
public class FileOperations {

    // variable declaration
    static Scanner scan = new Scanner(System.in);
    public final static String path = "D:\\Phase-1_Project\\Files";

    /**
     * Method Name: addFile()
     * This function is used to add user defined file in directory
     * Inside method:
     * 1.Take file name from user
     * 2.Check if file not exist create file
     * 3.Ask user for number for line needed to write
     * 4.write user provide lines in file
     */
    public static void addFile(){
        File file = new File(path);
        List<String> list = Arrays.asList(file.list());
        System.out.println("Enter name for file");
        String fileName = scan.nextLine();
        if(!(list.contains(fileName+".txt"))){
            System.out.println("How may line want to write?");
            int lineCount = Integer.parseInt(scan.nextLine());
            System.out.println("Enter file content here");
            try {
                FileWriter fr = new FileWriter(path+"\\ "+fileName+".txt");
                for(int i =0; i< lineCount;i++) {
                    fr.write(scan.nextLine());
                    fr.write('\n');
                }
                fr.close();
            }
            catch (IOException e) {
                e.printStackTrace();
            }
            System.out.println("File added successfully.\n");
        }
        else
            System.out.println("file name already exists in directory.");
    }

    /**
     * Method Name: deleteFile()
     * this method is used to delete file exist in main directory
     * Inside method:
     * 1. Asked user for file name to delete
     * 2. Check for file is it exist in directory
     * 3. If file exist delete file otherwise show message "file not found"
     */
    public static void deleteFile(){
        File file = new File(path);
        List<String> list = Arrays.asList(file.list());
        System.out.println("Enter name of file need to delete");
        String fileName = scan.nextLine()+".txt";
        if((list.contains(fileName))){
            File filedelete = new File(path+"\\ "+fileName);
            filedelete.delete();
            System.out.println(filedelete.getName()+" is deleted successfully\n");
        }
        else
    
```

```

        {
            System.out.println("File not found");
        }
    }

    /**
     * Method Name: searchFile()
     * This method is used to search file from main directory
     * Inside method:
     * 1. Asked user for file name to search in main directory
     * 2. Check if file exist show file path otherwise show message 'file not found'
     */
    public static void searchFile(){
        File file = new File(path);
        List<String> list = Arrays.asList(file.list());
        System.out.println("Enter name of file need to search");
        String fileName = scan.nextLine();
        if(list.contains(fileName+".txt")) {
            System.out.println("File exit at "+file.getAbsolutePath()+"\\"+fileName+".txt");
        }
        else
        {
            System.out.println("file not found");
        }
    }
}

```

➤ LockedMe.java

```

package com.Lockers.LockedMe;

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

public class LockedMe {

    public final static String path = "D:\\Phase-1_Project\\Files";

    /**
     * Method Name: displayMainMenu()
     * This method print welcome screen and main menu of LockedMe.com
     * Menu as below:
     * 1. Retrieving the file names in an ascending order
     * 2. Business-level operations - add, delete, search files
     * 3. Option to close application
     */
    public static void displayMainMenu() {
        System.out.println(":::*****:");
        System.out.println("  Welcome to LockedMe.com  ");
        System.out.println("      developed by Manvendra  ");
        System.out.println(":::*****:");
        System.out.println("1. Retrieve all files");
        System.out.println("2. File operation - delete, add or search files");
        System.out.println("3. Exit application");
    }

    /**
     * Method Name: retrieveAllFiles()
     * This function retrieve all files in main directory in ascending order

```

```

* {@value #path} : it contain main directory path
*/

public static void retrieveAllFiles() {
    File file = new File(path);
    if(file.exists()) {
        List<String> list = Arrays.asList(file.list());
        Collections.sort(list);
        for(String v : list) {
            System.out.println(v);
        }
    }
    else {
        System.out.println("No file exit.");
    }
}

public static void fileOperations(Scanner scan) {
    //Scanner scan = new Scanner(System.in);
    boolean run = true ;
    do {
        try {
            System.out.println("Choose operation need to perform from below options");

            System.out.println("=====");
            System.out.println("1. Add file");
            System.out.println("2. Delete file");
            System.out.println("3. Search file");
            System.out.println("4. Return to main menu");

            System.out.println("=====");
            int choice = Integer.parseInt(scan.nextLine());
            switch(choice) {
                case 1 -> FileOperations.addFile();
                case 2 -> FileOperations.deleteFile();
                case 3 -> FileOperations.searchFile();
                case 4 -> run = false;
                default -> System.out.println("Invalid input");
            }
        } catch (NumberFormatException e) {
            e.printStackTrace();
        }
    }while(run == true);
}

/**
 * Method Name : exit()
 * This method is used to exit from application.
 */
public static void exit() {
    System.exit(0);
}
}

```

Project Screen Shot

➤ Welcome Screen and main menu

```
:::::*****:::  
      Welcome to LockedMe.com  
              developed by Manvendra  
:::::*****:::  
1. Retrieve all files  
2. File operation - delete, add or search files  
3. Exit application  
Please enter your choice  
C:\Users\manve\Desktop  
1
```

➤ Option-1 : Retrieve all files

```
:::::*****:::  
      Welcome to LockedMe.com  
              developed by Manvendra  
:::::*****:::  
1. Retrieve all files  
2. File operation - delete, add or search files  
3. Exit application  
Please enter your choice  
1  
abc.txt  
filea.txt  
fileb.txt  
filec.txt  
filed.txt
```


➤ Option-2.1: File Operations – Add File

```
:::::*****:
Welcome to LockedMe.com
          developed by Manvendra
:::::*****:
1. Retrieve all files
2. File operation - delete, add or search files
3. Exit application
Please enter your choice
2
Choose operation need to perform from below options
=====
1. Add file
2. Delete file
3. Search file
4. Return to main menu
=====
1
Enter name for file
myFile
How may line want to write?
2
Enter file content here
I am Manvendra
Learning from Simplilearn
File added successfully.
```

➤ Option-2.2: File Operations – Delete File

```
:::::*****:
Welcome to LockedMe.com
          developed by Manvendra
:::::*****:
1. Retrieve all files
2. File operation - delete, add or search files
3. Exit application
Please enter your choice
2
Choose operation need to perform from below options
=====
1. Add file
2. Delete file
3. Search file
4. Return to main menu
=====
2
Enter name of file need to delete
myFile
myFile.txt is deleted successfully
```

➤ Option-2.3: File Operations – Search File

```
:::::*****:
Welcome to LockedMe.com
        developed by Manvendra
:::::*****:
1. Retrieve all files
2. File operation - delete, add or search files
3. Exit application
Please enter your choice
2
Choose operation need to perform from below options
=====
1. Add file
2. Delete file
3. Search file
4. Return to main menu
=====
3
Enter name of file need to search
abc
File exit at D:\Phase-1_Project\Files\abc.txt
```

➤ Option-2.4: Return to main menu

```
:::::*****:
Welcome to LockedMe.com
        developed by Manvendra
:::::*****:
1. Retrieve all files
2. File operation - delete, add or search files
3. Exit application
Please enter your choice
2
Choose operation need to perform from below options
=====
1. Add file
2. Delete file
3. Search file
4. Return to main menu
=====
4
:::::*****:
Welcome to LockedMe.com
        developed by Manvendra
:::::*****:
1. Retrieve all files
2. File operation - delete, add or search files
3. Exit application
Please enter your choice
```

➤ **Option-3: Exit application**

```
:::*****:::
Welcome to LockedMe.com
          developed by Manvendra
:::*****:::
1. Retrieve all files
2. File operation - delete, add or search files
3. Exit application
Please enter your choice
3
```

Git Repository URL

- [https://github.com/Manvendra1097-eng/Phase-1 Project-LockedMe.com.git](https://github.com/Manvendra1097-eng/Phase-1_Project-LockedMe.com.git)