SOURCE CODE

LockedMe.com
Secure App

SOURCE CODE

Main Class

```
package com.lockers;
import java.util.Scanner;

public class Main {
	private static Scanner sc = new Scanner(System.in);

	public static void main(String[] args) {

		System.out.println("\n*******************************************************\n\n\n");
		System.out.println("\tWelcome to LockeMe.com secure app\n\n\n\n\n");
		System.out.println("\tDeveloped by: Dakamanbha Ryngkhlem\n" + "\n\tSeptember, 2022");
		System.out.println("\n*******************************************************\n");

		displayInfo();

		boolean quit = false;
		while(!quit) {
			System.out.println("\nEnter your choice: ");
			int choice = sc.nextInt();
			switch(choice) {

				case 1:
					UserInfo.userInteraction();
					break;
				case 2:
					quit=true;
					System.out.println("Exiting from the application...");
					break;
				default:
					System.out.println("Please enter only numbers from 0-1");
					break;
			}
		}
	}

	public static void displayInfo() {
		System.out.println("Press #1 To continue");
		System.out.println("Press #2 To exit");
	}
}
```

User Interaction Information

```
package com.lockers;
import java.io.File;
import java.io.FileFilter;
import java.util.Arrays;
import java.util.Comparator;
import java.util.Scanner;

public class UserInfo{

	private static Scanner sc = new Scanner(System.in);

	public static void userInteraction() {

		boolean quit = false;
		int choice = 0;

		printInstructions();

		 while(!quit){
			System.out.println("Enter the action to perform in User Interaction: "
					+ "(1 - Display Instructions) ");
			choice = sc.nextInt();

			switch(choice) {
				case 1:
					UserInfo.printInstructions();
					break;
				case 2:
					sortFiles();
					break;
				case 3:
					BusinessLevelOperations.operation();
					break;
				case 0:
					quit = true;
					Main.displayInfo();
					break;
				default:
					System.out.println("Please enter only numbers from 0-3");
					break;
			}
		}
	}

	private static void printInstructions() {
		System.out.println("User Interaction Information");
		System.out.println("\t 1 - Display all instructions");
		System.out.println("\t 2 - To retrieve the file names in an ascending order");
		System.out.println("\t 3 - To enter to Business-level operations");
		System.out.println("\t 0 - To go back to the main screen");
	}

	private static void sortFiles() {

		Scanner sc = new Scanner(System.in);
		System.out.println("Enter the file path: ");
	   String dirPath = sc.nextLine(); // Takes the directory path as the user input

	   File folder = new File(dirPath);
	   if(folder.isDirectory()) {

		File[] fileList = folder.listFiles();

		Arrays.sort(fileList);
```

```
            System.out.println("\nTotal number of items present in the directory: " + fileList.length );

            System.out.println("\nPrinting all available files in the directory");

            // Lists only files since we have applied file filter
            for(File file:fileList) {
               System.out.println("\t" +file.getName());
            }
            System.out.println("----------------------------------------\n");

            // Creating a filter to return only files.
            FileFilter fileFilter = new FileFilter()
            {
               @Override
               public boolean accept(File file) {
                  return !file.isDirectory();
               }
            };

            fileList = folder.listFiles(fileFilter);

            // Sort files in ascending order alphabetically
            Arrays.sort(fileList, new Comparator<Object>()
            {
               @Override
               public int compare(Object f1, Object f2) {
                  return ((File) f1).getName().compareToIgnoreCase(((File) f2).getName());
               }
            });
            System.out.println("Printing only files in ascending order\n ");
            //Prints the files in file name ascending order
            for(File file:fileList)
            {
                     System.out.println(file.getName());
            }
            System.out.println("**********************************************");
         }
          else {
                  System.out.println("Wrong Directory Path or Path doesn't exists");
         }
      }
}
```

Business Level Operations

```
package com.lockers;
import java.io.File;
import java.io.IOException;
import java.util.Scanner;

public class BusinessLevelOperations{

	private static Scanner sc = new Scanner(System.in);

	public static void operation() {
		printInstructions();

		boolean quit = false;
		int choice = 1;
		while(!quit) {
			System.out.println("Enter the operations you would like to perform in Business Operation: ");
			choice = sc.nextInt();
			switch(choice) {
			case 0:
				quit=true;
				break;
			case 1:
				addfile();
				break;
			case 2:
				deleteFile();
				break;
			case 3:
				searchForFile();
				break;
			default:
				System.out.println("Please enter only numbers from 0-3");
				break;
			}
		}
	}
	private static void printInstructions() {
		System.out.println("Entering Business Level-Operations"
		+"\n 1 - To add a user specified file to the application"
		+"\n 2 - To delete a user specified file from the application"
		+"\n 3 - To search a user specified file from the application"
		+"\n 0 - To go back to the User Interface Interaction");
	}
	private static void searchForFile() {
		System.out.println("Enter your file name you would like to search: ");
		String inputFile = sc.next();
		sc.nextLine();
		File searchFile = new File(inputFile);
			if(searchFile.exists()){
				System.out.println("\n"+searchFile.getName()+" found. " );
			}
			else{
				System.out.println("\n" +searchFile.getName()+" file is not found or file does not exist");
			}
	}
```

```
	private static void deleteFile() {
		System.out.println("Enter your file name you would like to delete: ");
		String filename = sc.next();
		File myfile = new File(filename);
		if(myfile.delete()) {
			System.out.println("\n"+myfile.getName() +" is deleted successfully ");
		}
		else {
			System.out.println("\n"+myfile.getName() +" failed to be deleted or file does not exist");
		}
	}
	private static void addfile() {
		try {
			System.out.println("Enter your file name you would like to create: ");
			String filename = sc.next();

			File myObj = new File(filename);
			if(myObj.createNewFile()) {
				System.out.println("\n" +filename+ " File Created Successfully");
			}
			else {
				System.out.println("\nFile already exist");
			}
		}catch (IOException e) {
			System.out.println("\nAn error occured");
			e.printStackTrace();
		}
	}
}
```
