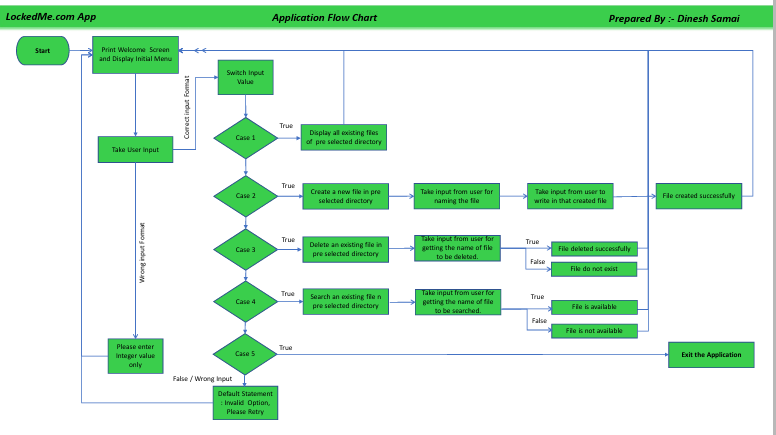
## **The code for this project is hosted at**

<https://github.com/Dinesh3108/LockedMe_Application>

The project is developed by Dinesh Samai (Dinesh3108)

## **Core concepts used in project**

* Collections framework.
* Conditional Statements.
* Loop concept.
* File Handling
* Sorting
* Exception Handling
* **Flow of the Application**



Click on given link to go to open flow chart[1. Flow of Application.pdf](1.%20Flow%20of%20Application.pdf)

* **Demonstrating the product capabilities, appearance, and user interactions.**

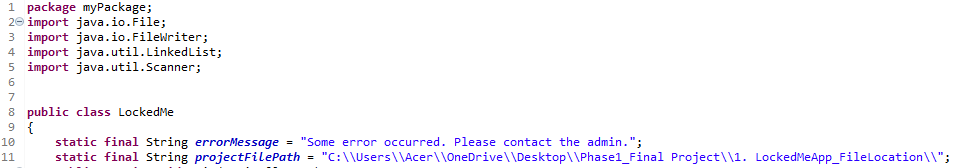
To demonstrate the product capabilities, below are the sub-sections configured to highlight appearance and user interactions for the project:

1. [Creating the project in Eclipse.](#Creatingaprojectineclipse)
2. [Specifying the path of project folder where file operations will be done.](#Specifyingthepathprojectfolder)
3. [Creating a switch case for all application menus inside a loop of Main method.](#Switchcase)
4. [Creating a method for showing the welcome screen & menu.](#Welcomescreen)
5. [Creating a method for display all existing files in the directory.](#Displayfiles)
6. [Creating a method for creating a new file in the directory.](#CreateFiles)
7. [Creating a method for deleting file in the directory.](#DeleteFiles)
8. [Creating a method for searching a file in the directory.](#SearchFiles)

**Step 1:** **Creating a new project in Eclipse**

1. Open Eclipse
2. Go to File -> New -> Project -> Java Project -> Next.
3. Type in any “LockedMeApp” in project name and click on “Finish.”
4. Select your project -> Src -> New -> Package -> Type in “myPackage” and click on “Finish.”
5. Select your project and go to File -> New -> Class.
6. Enter LockedMe in class name, check the checkbox “public static void main(String[] args)”, and click on “Finish.”

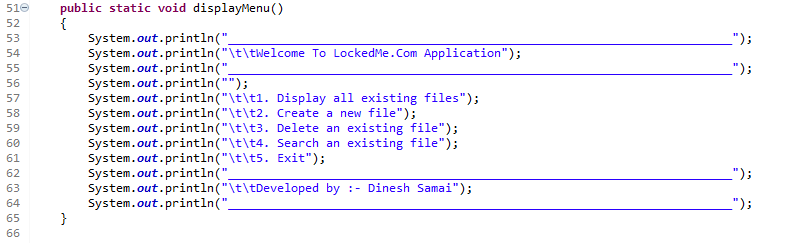
**Step 2:** **Specifying the path of project folder where file operations will be done.**



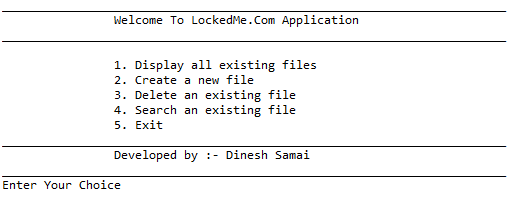
**Step 3:** **Creating a switch case for all application menu inside a loop of Main method.**



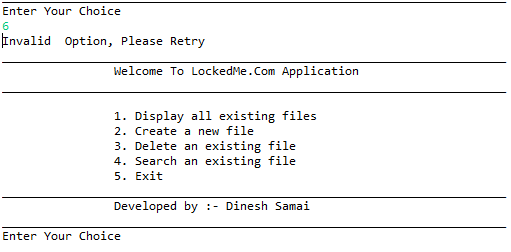
**Step 4:** **Creating a method for showing the welcome screen & menu.**



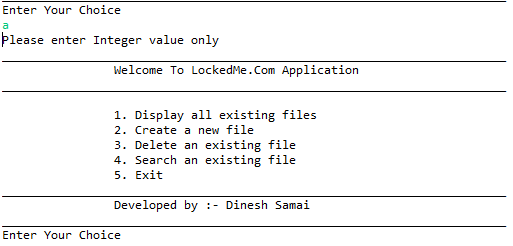
**Output: Welcome Screen & Menu Options.**



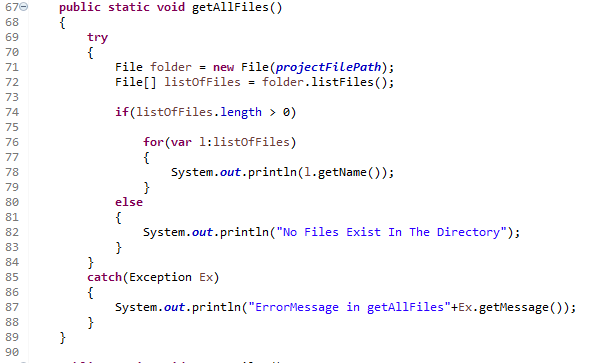
**Output: If user enters a invalid option (integer type).**



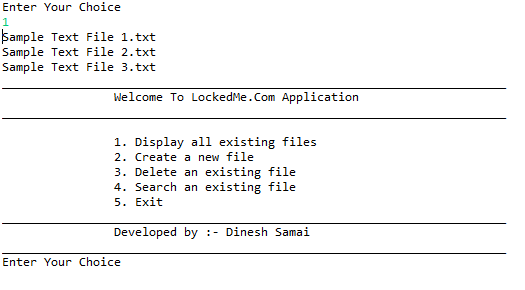
**Output: If user enters an invalid option (string type).**



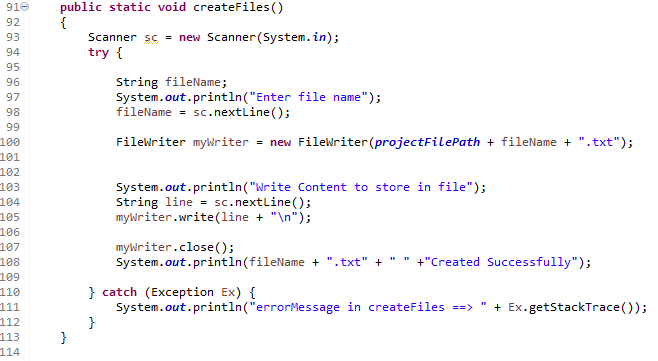
**Step 5:** **Creating a method for display all existing files in the directory.**



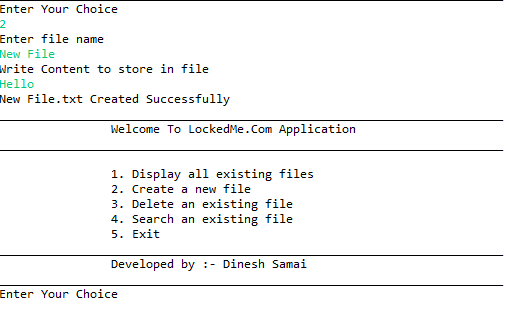
**Output:**



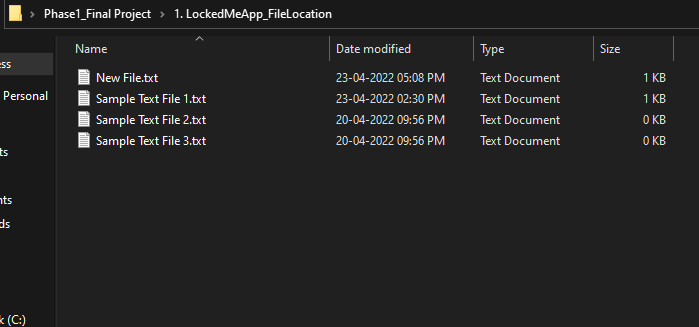
**Step 6:** **Creating a method for creating a new file in the directory.**



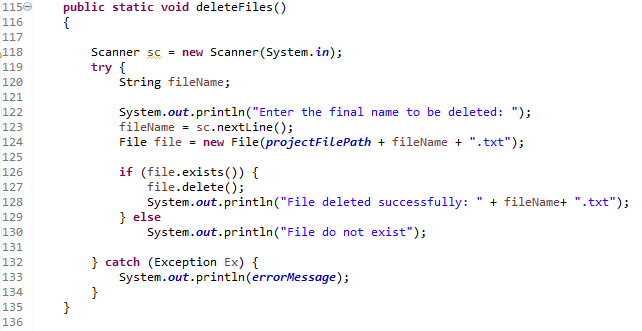
**Output:**



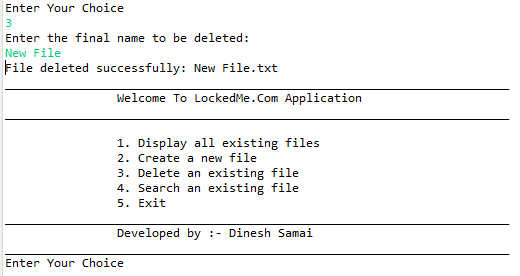
**Showing the newly created file in the directory.**



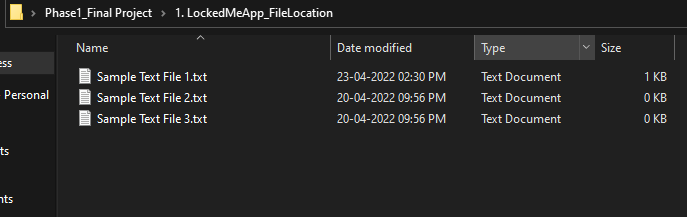
**Step 7:** **Creating a method for deleting file in the directory.**



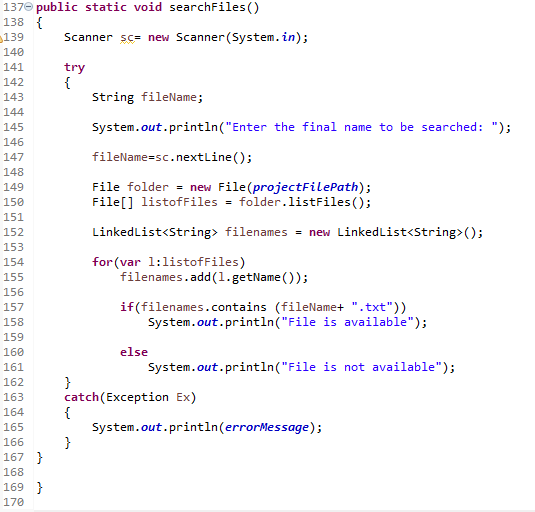
**Output:**



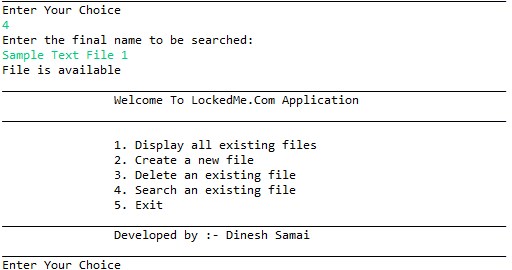
**Showing that the file does not exist anymore.**



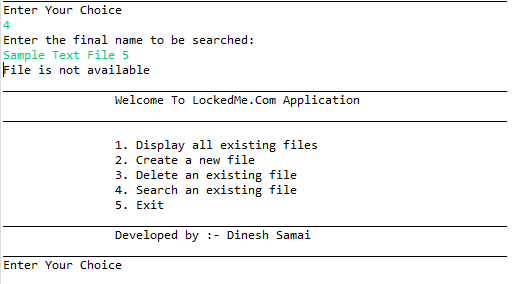
**Step 8:** **Creating a method for searching a file in the directory.**



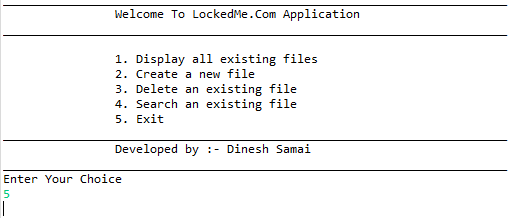
**Output: If File Exists**



**Output: If File do not Exists**



**Output: Exit the Application.**



**Unique Selling Points of the Application**

* 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.
* User is also provided the option to write content if they want into the newly created file.
* The format (.txt) is a default feature. User does not have to write the file format while creating, deleting & searching any file.