```
Package VirtualKey;
        Import java.io.File;
        Import java.io.IOException;
        Import java.util.Arrays;
        Import java.util.Scanner;

        Public class LockedMe {
          Static String DIRECTORY;
          File folder_name;

          Public LockedMe() {
            DIRECTORY = System.getProperty("user.dir");
            Folder_name = new File(DIRECTORY+"/files");
            If (!folder_name.exists())
              Folder_name.mkdirs();
            System.out.println("DIRECTORY : "+ folder_name.getAbsolutePath());
          }

          Private static final String WELCOME_PROMPT =
              "\n**********  LockedMe.com **********"+
                  "\n********** Sathvika Munja **********\n";

          Private static final String MAIN_MENU_PROMPT =
              "\nMAIN MENU – Select any of the following: \n"+
                  "1 -> List files in directory\n"+
                  "2 -> Add, Delete or Search\n"+
                  "3 -> Exit Program";

          Private static final String SECONDARY_MENU_PROMPT =
```

```
    “  \nSelect any of the following: \n”+
        “  a -> Add a file\n”+
        “  b -> Delete a file\n”+
        “  c -> Search a file\n”+
        “  d -> GoBack”;

Void showPrimaryMenu() {
  System.out.println(MAIN_MENU_PROMPT);
  Try{
    Scanner scanner = new Scanner(System.in);
    Int option = scanner.nextInt();
    Switch (option){
      Case 1 : {
        showFiles();
        showPrimaryMenu();
      }
      Case 2 : {
        showSecondaryMenu();
      }
      Case 3 : {
        System.out.println(“Thank You”);
        System.exit(0);
      }
      Default: showPrimaryMenu();
    }
  }
  Catch (Exception e){
    System.out.println(“Please enter 1, 2 or 3”);
    showPrimaryMenu();
```

```
    }
}

Void showSecondaryMenu() {
    System.out.println(SECONDARY_MENU_PROMPT);
    Try{
        Scanner scanner = new Scanner(System.in);
        Char[] input = scanner.nextLine().toLowerCase().trim().toCharArray();
        Char option = input[0];

        Switch (option){
            Case ‘a’ : {
                System.out.print(“↳ Adding a file…Please Enter a File Name : “);
                String filename = scanner.next().trim().toLowerCase();
                addFile(filename);
                break;
            }
            Case ‘b’ : {
                System.out.print(“↳ Deleting a file…Please Enter a File Name : “);
                String filename = scanner.next().trim();
                deleteFile(filename);
                break;
            }
            Case ‘c’ : {
                System.out.print(“↳ Searching a file…Please Enter a File Name : “);
                String filename = scanner.next().trim();
                searchFile(filename);
                break;
            }
```

```
        Case ‘d’ : {
            System.out.println(“Going Back to MAIN menu”);
            showPrimaryMenu();
            break;
        }
        Default : System.out.println(“Please enter a, b, c or d”);
      }
      showSecondaryMenu();
   }
   Catch (Exception e){
      System.out.println(“Please enter a, b, c or d”);
      showSecondaryMenu();
   }
}

Void showFiles() {
   If (folder_name.list().length==0)
      System.out.println(“The folder is empty”);
   Else {
      String[] list = folder_name.list();
      System.out.println(“The files in “+ folder_name +” are :”);
      Arrays.sort(list);
      For (String str:list) {
         System.out.println(str);
      }
   }
}

Void addFile(String filename) throws IOException {
```

```
   File filepath = new File(folder_name +"/"+filename);
   String[] list = folder_name.list();
   For (String file: list) {
      If (filename.equalsIgnoreCase(file)) {
         System.out.println("File " + filename + " already exists at " + folder_name);
         Return;
      }
   }
   Filepath.createNewFile();
   System.out.println("File "+filename+" added to "+ folder_name);
}

Void deleteFile(String filename) {
   File filepath = new File(folder_name +"/"+filename);
   String[] list = folder_name.list();
   For (String file: list) {
      If (filename.equals(file) && filepath.delete()) {
         System.out.println("File " + filename + " deleted from " + folder_name);
         Return;
      }
   }
   System.out.println("Delete Operation failed. FILE NOT FOUND");
}

Void searchFile(String filename) {
   String[] list = folder_name.list();
   For (String file: list) {
      If (filename.equals(file)) {
         System.out.println("FOUND : File " + filename + " exists at " + folder_name);
```

```
            Return;
          }
        }
        System.out.println("File NOT found (FNF)");
    }

    Public static void main(String[] args) {
        System.out.println(WELCOME_PROMPT);
        LockedMe menu = new LockedMe();
        Menu.showPrimaryMenu();
    }
}
```