# Lockedme.java

|  |
| --- |
| package *com*;  *public* *class* lockedme {  *public* *static* void main(String[] args) {  *//From the main app, we launch the three functions from the menu*  *//To show the welcome message, the main menu options, and the function to navigate the menu*          Menu.showWelcome();          Menu.navigationMainMenu();      }  } |

# Menu.java

|  |
| --- |
| package *com*;  import *java.util.InputMismatchException*;  import *java.util.Scanner*;  *public* *class* Menu {  *//Function with the welcome message*  *public* *static* void showWelcome() {          System.out.println("--------------------");          System.out.println("    LockedMe.com    ");          System.out.println("--------------------");          System.out.println("Application developed:\n"                  + "    by Francisco García (franciscogarciapozo@gmail.com");          System.out.println("--------------------");      }    *//Function with the messages from the main menu*  *public* *static* void showMainMenu() {          System.out.println("With this application you can work with the files in the folder X");          System.out.println("Choose one of the following opions:");          System.out.println("1. List the files in the folder");          System.out.println("2. Operate with files in the folder ");          System.out.println("3. Close the application");          System.out.println("Insert you option: ");      }    *//Function with the messages from the file submenu*  *public* *static* void showFilesSubmenu() {          System.out.println("What operation do you want to do with the files in the folder?:");          System.out.println("Choose one of the following opions:");          System.out.println("1. Create a new file in the folder");          System.out.println("2. Delete a file from the folder");          System.out.println("3. Search for a file in the folder");          System.out.println("4. Back to the previous menu");          System.out.println("Insert you option: ");      }    *//Function with the logic of the main menu*  *public* *static* void navigationMainMenu() {  *//infinite loop. We don't leave the application until the user insert 3*          while(true) {              showMainMenu();              int optionSelected = optionSelection();  *//While the user doesn't insert a correct option, the loop continues*              while (optionSelected <= 0 || optionSelected > 3) {                  System.out.println("Introduce a number between 1 and 3. ");                  optionSelected = optionSelection();              }  *//Depends of the option selected, the different functions are called, of the application is closed with the case 3*              switch (optionSelected) {              case 1:                  FilesOptions.listFilesInFolder();                  break;              case 2:                  navigationFilesMenu();                  break;              case 3:                  System.exit(0);                  break;              }          }      }      *public* *static* void navigationFilesMenu() {  *//infinite loop. We don't leave the application untin the user insert 4*          while(true) {              showFilesSubmenu();              int optionSelected = optionSelection();  *//While the user doesn't insert a correct option, the loop continues*              while (optionSelected <= 0 || optionSelected > 4) {                  System.out.println("Incorrect option: You need to introduce a number between 1 and 4");                  optionSelected = optionSelection();              }  *//Depends of the option selected, the different functions are called*              switch (optionSelected) {              case 1:                  FilesOptions.createFile();                  break;              case 2:                  FilesOptions.deleteFile();                  break;              case 3:                  FilesOptions.searchFile();                  break;              case 4:                  navigationMainMenu();                  break;              }          }      }    *//Function to read the option from the user. A exception for another type of input is caught*  *public* *static* int optionSelection() {          int optionSelected = 0;          Scanner sc = new Scanner(System.in);          try {              optionSelected = sc.nextInt();          } catch(InputMismatchException e) {              System.out.println("You MUST introduce a number. ");          }          return optionSelected;      }  } |

# FilesOptions.java

|  |
| --- |
| package *com*;  import *java.io.File*;  import *java.io.IOException*;  import *java.util.Arrays*;  import *java.util.Scanner*;  *public* *class* FilesOptions {  *//Constant with the name of the working directory*  *static* *final* String WORK\_DIRECTORY = "working directory";    *//Function to list all the files in the working directory*  *public* *static* void listFilesInFolder() {  *//A File object with the working directory is created*          File fi = new File(WORK\_DIRECTORY);  *//With the function list(), we create a String array with all the files in the folder*          String[] arrayFilesName = fi.list();  *//If the array is empty (length 0), a message to the user explaining the situation is showed*          if(arrayFilesName.length == 0) {              System.out.println("-------------------------------");              System.out.println("The directory is empty");          } else {  *//The array is sorted with sort(array), and a for loop is used to show all the results in the array*              Arrays.sort(arrayFilesName);              System.out.println("Files in the working directory:");              System.out.println("-------------------------------");              for (String fileName: arrayFilesName) {                  System.out.println(fileName);              }          }          System.out.println("-------------------------------");      }    *//Function to create a file in the working directory*  *public* *static* void createFile() {          Scanner sc = new Scanner(System.in);          String fileName;            System.out.println("Insert the name of the file to create:");          fileName = sc.next();  *//A File object with the file name is created*          File fi = new File(WORK\_DIRECTORY + "\\" + fileName);  *//With File.exists() the existence of the file is checked. If the file already exists, a message is showed to the user*          if(fi.exists()) {              System.out.printf("The file %s can't be created because a file with this name exists in the working directory\n", fileName);          }          try {  *//Trying to create the file. Also a IOExcepcion is caught*              if(fi.createNewFile()) {                  System.out.printf("File %s created correctly\n", fileName);              } else {                  System.out.printf("The file %s can't be created\n", fileName);              }          } catch (IOException e) {              System.out.println("Error IO creating the file: " + e.toString());          }      }    *//Function to check if a file exists in the working direcotry*  *public* *static* void searchFile() {          Scanner sc = new Scanner(System.in);          String fileName;            System.out.println("Insert the text of the file to find:");          fileName = sc.next();  *//A File object with the file name is created*          File fi = new File(WORK\_DIRECTORY + "\\" + fileName);  *//With File.exists() the existence of the file is checked.*          if(fi.exists()) {              System.out.printf("The file %s exists in the working directory\n", fileName);          } else {              System.out.printf("The file %s doesn't exist in the working directory\n", fileName);          }      }    *//Function to delete a file from the working directory*  *public* *static* void deleteFile() {          Scanner sc = new Scanner(System.in);          String fileName;          System.out.println("Insert the name of the file to delete:");          fileName = sc.next();  *//A File object with the file name is created*          File fi = new File(WORK\_DIRECTORY + "\\" + fileName);  *//With File.exists() the existence of the file is checked. If the file already exists, a message is showed to the user*          if(fi.exists()) {              if(fi.delete()) {                  System.out.printf("File %s deleted correctly\n", fileName);              } else {                  System.out.printf("The file %s can't be deleted\n", fileName);              }          } else {              System.out.printf("The file %s doesn't exists\n", fileName);          }      }  } |

More info about the project and the source code can be found in the GitHub project:

<https://github.com/FranciscoGP-telco/LockedMe.com>