Create an Interface LockedMeScreen and create 3 abstract method

Create a class WelcomeScreen and implement 3 abstract class

Initailize Welcome Text, Developer Text and an ArrayList for Main Menu options

Write a Constructor for displaying Main Menu options using above declared ArrayList.

Write a method showWelcomeMessage() for displaying Welcome Text and Developer Text

Write a showMenu() method for displaying Main Menu on Welcome screen using ArrayList of String.

Write getInputFromUser() method to get input from user for Main Menu options.

In getInputFromUser() method, create a getNavigateOption()

A

A

Add one more method navigateOption(userInput) in getInputFromUser() method to get option from User using Switch Case. In Switch Case, write case 1 for displaying available file with showFiles() method, case 2 for navigating to File Operation Menu and default for invalid options from User.

Get input as integer with Scanner class in try block.In Exception Handling, handle InputMismatchException and IllegalArgumentException with catch block.

Create FileDirectory class in function package.Initailize static final String filepath as "src\\directory\\" and ArrayList of File.

Write getName() method that return filenames present at filepath, showFiles() method for displaying files and method getFiles() for adding list of files to ArrayList of files and for retrieving sorted files available in directory folder.

Create a class DirectoryFunctions in functions package and write a static method.

Write a showFiles() method in WelcomeScreen class which is used to show files when User enters Main Menu option as 1.

Create a class ScreenFunctions to switch between WelcomeScreen and FileOperationsScreen. LockedMeScreen interface that is assign to object of WelcomeScreen class.Also write Setter and Getter for this class.

Now create FileOperationsScreen class which implements LockedMeScreen interface in same ArrayList of String.

B

B

Save your project and commit code to repository

Write a method addFile() that gets input filename which User wants to Add in system.Create a object of File named as file in try block.Inside the try block put If condition as file.createNewFile(); and call fileDirectory.getFiles().add(file); method.If File Added Successfully it will return true in If condition else it will show message as"This File Already Exits, no need to add another".In catch block throw IOException.

Create a package lockedme and write a main method inside a class LockedMeApplication. Now initialize a object of Welcome Screen class and call showWelcomeMessage() method first and then getInputFromUser().

Write a method searchFile() that gets input filename which User wants to Search in system.In this method, create ArrayList of File which stores the list of available files in sytem.Using .equals method in If condition search a required filename and if required filename found in ArrayList of files, set Boolean variable found to True else set it to False.

Write a method deleteFile() that gets input filename which User wants to Delete in system.Initailize Path variable and create object of File to get filenames.Inside If condition, call fileDirectory.getFiles().remove(file); method to delete file.And in else part write "Failed to delete file, file was not found." message.