https://github.com/KayrXa/JavaDevelopmentProjects/blob/master/VirtualKeyForOurRepositories/src/LockedMe.java

import java.io.File;

import java.io.IOException;

import java.util.Scanner;

public class LockedMe {

// Using input fomr case 3 search file

// Search selected file into subdirs, if not added directory search in current, if not found

public static void findFile(String sname, File sfile){

File[] list = sfile.listFiles();

if (list!=null){

for (File fil : list)

{

if (fil.isDirectory()){

findFile(sname,fil);

} else if (sname.equalsIgnoreCase(fil.getName())) {

System.out.println("File "+ sname +" was found in path " + fil.getParentFile());

}

}

} else {

System.out.println("Sorry you put invalid input, try it again");

}

}

public static void baseWelcome() {

//Printing Welcome screen

System.out.println("Welcome to Locked page");

System.out.println("Application name: LockedMe4Ever");

System.out.println("Author: Pavel Rygl");

System.out.println("Simple Learn ID: ");

}

public static void mainMenu() {

//Printing basic menu

System.out.println();

System.out.println("Menu");

System.out.println("0. Close application");

System.out.println("1. Show current file names in ascending order");

System.out.println("2. Modify file");

System.out.println();

System.out.println("Please select one of the following choices \"0\" \"1\" \"2\" ");

System.out.println("----------------");

}

public static void subMenu() {

System.out.println();

System.out.println("Modifying menu");

System.out.println("1) Add a file to existing directory list");

System.out.println("2) Delete file from existing directory list");

System.out.println("3) Search for the file from the main directory");

System.out.println("4) Back to main menu");

System.out.println("5) Close application");

System.out.println();

System.out.println("Please select one of the following choices \"1\" \"2\" \"3\" \"4\" \"5\" ");

System.out.println("----------------");

}

public static void processingMenu() throws IOException {

int choice;

//

while (true) {

Scanner sc = new Scanner(System.in).useDelimiter("\n");

mainMenu();

try {

choice = sc.nextInt();

} catch (Exception e) {

System.out.println("Input is not valid, please try it again");

choice = 3;

}

// Getting input of 1st menu

switch (choice) {

// OK

case 0 -> {

System.out.println("Closing application");

System.out.println("See you!");

System.out.println("----------------");

System.exit(0);

}

case 1 -> {

// Calls method - Return files in ascending order, if not selected choose form current, if it is empty / not exist

System.out.println("Please enter directory you want to list, otherwise it will print current root directory");

String dirname = sc.next();

if ("".equals(dirname)) {

dirname = "user.dir";

System.out.println("Printing files in selected folder " +dirname+ " in ascending order");

// !! resolve issue if any other dir, or when not inserted anything now just working when typing user.dir

String path = System.getProperty(dirname);

File files = new File(path);

System.out.println(dirname);

for (File f:files.listFiles()) {

if (f.isFile())

System.out.println(f);

}

} else {

System.out.println("You entered " +dirname+ " which is not a dir");

}

System.out.println("----------------");

}

// OK

case 2 -> {

System.out.println("Transferring to submenu");

System.out.println("----------------");

processingSubMenu();

}

case 3 -> {

System.out.println("Refreshed");

processingMenu();

}

}

}

}

public static void processingSubMenu() throws IOException {

int choicesub;

//

loop: while (true) {

Scanner sc = new Scanner(System.in).useDelimiter("\n");

subMenu();

try {

choicesub = sc.nextInt();

} catch (Exception e) {

System.out.println("Input is not valid, please try it again");

choicesub = 6;

}

switch (choicesub) {

/\*

loop: while (true) {

subMenu();

Scanner sc = new Scanner(System.in).useDelimiter("\n");

int choicesub = sc.nextInt();

switch (choicesub) {

\*/

case 1:

//OK Catching file to create

System.out.println("Please write down file you want to create");

String nfile = sc.next();

// OK Catching path to create

// !! Format: C:\\Users\\austr\\IdeaProjects\\Projects\\VirtualKeyForOurRepositories\\Programoutput

System.out.println("Ok, preparing to create file " + nfile + " Now type a directory subdirectory where you want to put it, if in current hit enter");

Scanner pth = new Scanner(System.in).useDelimiter("\n");

String path = pth.next();

if ("".equals(path)) {

path = "C:\\Users\\austr\\IdeaProjects\\Projects\\VirtualKeyForOurRepositories\\Programoutput";

}

//full path: C:\\Users\\austr\\IdeaProjects\\Projects\\VirtualKeyForOurRepositories\\Programoutput

//Initializing file for dir

File file = new File(path, nfile);

//Trying to create file, check if exists, if not then create. If non dir is inserted then throw exception and menu again

try {

if (file.exists() && file.isFile()) {

System.out.println("Sorry file was not created");

System.out.println(nfile + " already exist in " + path);

} else {

System.out.println("Trying to create file " + nfile + " in directory " + path);

file.createNewFile();

if (file.exists() && file.isFile()){

System.out.println("File " +nfile + " was successfully created");

}else

System.out.println("Sorry something went wrong, try it again or contact administrator");

}

}catch (IOException c) {

System.out.println("Sorry " + path + " is not a directory");

}

System.out.println("----------------");

break;

case 2:

// OK delete selected file from chosen directory, if not added directory delete form current, if not file or directory exist - throw exception, validation - case sensitive - message after action - deleted/not found

System.out.println("Type a absolute address of file you want to delete");

//Scan input

Scanner df = new Scanner(System.in).useDelimiter("\n");

String delpath = df.next();

//create object for delete file

File delefile = new File(delpath);

// Variant if not inseret anything, try to delete for current directory

if ("".equals(delpath)){

delpath = "C:\\Users\\austr\\IdeaProjects\\Projects\\VirtualKeyForOurRepositories\\Programoutput\\";

System.out.println("Ok we will delete file from current dir, but you must type name at least");

Scanner fil = new Scanner(System.in).useDelimiter("\n");

String filedel = fil.next();

File filedels = new File(delpath+filedel);

if (filedels.isFile()){

System.out.println("Deleting file " +filedels);

filedels.delete();

String checkdeletion = (filedels.exists()) ? "Something went wrong file was not deleted, please try again or contact administrator." : "File was successfully deleted!";

System.out.println(checkdeletion);

} else {

System.out.println("Sorry you put invalid input, try it again");

}

//will delete if have absolute path

//example C:\\Users\\austr\\IdeaProjects\\Projects\\VirtualKeyForOurRepositories\\Programoutput\\test.txt

} else {

if (delefile.isFile()){

System.out.println("Deleting file " +delefile);

delefile.delete();

String checkdeletion = (delefile.exists()) ? "Something went wrong file was not deleted, please try again or contact administrator." : "File was successfully deleted!";

System.out.println(checkdeletion);

} else {

System.out.println("Sorry you put invalid input, try it again");

}

}

System.out.println("----------------");

break;

case 3:

// 3. Scanner input for search file, cals findFile method

Scanner scan = new Scanner(System.in);

System.out.println("Type file you want to search:");

String searchedFile = scan.next();

System.out.println("Enter the directory where to search");

String searchedDirectory = scan.next();

findFile(searchedFile, new File(searchedDirectory));

System.out.println("----------------");

break;

case 4:

//OK Ends loop, go back to Main menu

System.out.println("Going back to main menu");

System.out.println("----------------");

break loop;

case 5:

//OK Finish program

System.out.println("Closing application");

System.out.println("See you!");

System.out.println("----------------");

System.exit(0);

break;

case 6:

System.out.println("Refreshed");

processingSubMenu();

}

}

}

public static void main(String[] args) throws IOException {

baseWelcome();

processingMenu();

}

}