// Yousif Al-Dhfeery

package prototype\_1;

import java.io.File;

import java.io.FileWriter;

import java.util.Scanner;

public class RVK {

public RVK() {

welcomeMsg(); // Print the welcome massage

mainOptions(); // // Print the Main Options

}

public static void welcomeMsg() { // Print the welcome massage

System.out.println("##########################################\n\n"

+ " --- Repositories Virtual Key ---\n\n"

+ "$- Devaloped by: Yousif Al-Dhfeery\n"

+ "$- https://github.com/xxmelar-ksaxx\n\n"

+ "##########################################");

}

public static void mainOptions() { // Print the Main Options

System.out.print("-----------------------------"

+ "\n -- Main Options Menu--\n"

+ "-----------------------------\n"

+ " 1- Display file names\n"

+ " 2- File Manipulation Options\n"

+ " 3- Exit\n\n"

+ " Plese type one option No: ");

MOSelector(); // Main options list

}

public static void filesMOptions() { // Print the File Manipulation Options

System.out.print("-----------------------------"

+ "\n -- File Manipulation Options -- \n"

+ "-----------------------------\n"

+ " 1- Add file\n"

+ " 2- Delete file\n"

+ " 3- Search for a file\n"

+ " 4- Return to Main Options Menu\n\n"

+ " Plese type one option No: ");

FMOSelector(); // Files Manipulation Options list

}

public static void MOSelector() { // Main options list

Scanner kb = new Scanner(System.in);

int x=0;

try {

x = kb.nextInt();

if( x<1 || x>3 ) { // The range of options from 1 to 3

System.out.println("Error!!. Wrong input. Plese try agin..");

mainOptions();

}

} catch (Exception e) {

System.out.println("Error!!. Wrong input. Plese try agin..");

mainOptions();

}

switch(x) {

case 1: // List the files in the directory

System.out.println("$-- Show the Files:");

showFiles(1); // set to '1' for Listing the files option

mainOptions();

break;

case 2: // Show File Manipulation Options

System.out.println("$-- Show File Manipulation Options");

filesMOptions();

mainOptions();

break;

case 3: // Exit the program

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

System.exit(0);

break;

}

}

public static void FMOSelector() { // Files Manipulation Options list

Scanner kb = new Scanner(System.in);

int x=0;

try {

x = kb.nextInt();

if( x<1 || x>4 ) { // The range of options from 1 to 4

System.out.print("Error!!. Wrong input. Plese try agin..");

filesMOptions();

}

} catch (Exception e) {

System.out.println("Error!!. Wrong input. Plese try agin..");

filesMOptions();

}

switch(x) {

case 1: // Create new file or folder

System.out.println("$-- Add new file");

addFile();

filesMOptions();

break;

case 2: // delete file or folder

System.out.println("$-- Delete a file");

deleteFile();

filesMOptions();

break; // Search for a file or folder

case 3:

System.out.println("$-- Serch for a file by name ");

showFiles(2); // set to '2' for searching option

filesMOptions();

break;

case 4: // Return to main options list

System.out.println("Return to main options");

mainOptions();

break;

}

}

public static void addFile() { // Create new file of folder

System.out.print(" Enter '1' for File, or '2' for Folder: ");

Scanner kb = new Scanner(System.in);

int x=0;

try {

x = kb.nextInt();

if( x<1 || x>2 ) { // The range of file type. 1 or 2

System.out.println("Error!!. Wrong input. Plese try agin..");

addFile();

}

} catch (Exception e) {

System.out.println("Error!!. Wrong input. Plese try agin..");

addFile();

}

switch(x) {

case 1: // To create a file

System.out.print("Enter file name: ");

String userInput="";

try {

userInput = kb.next();

} catch (Exception e) {

System.out.println("Error!!. Wrong input. Plese try agin..");

}

try {

FileWriter f = new FileWriter("DemoRepository\\"+userInput); // Create a file

f.close();

System.out.println(" Seccessfil: File created seccessfully");

} catch (Exception e) {

System.out.println(" Failed : file not created !!");

}

break;

case 2: // to create a folder

System.out.print("Enter folder name: ");

String userInput2="";

try {

userInput2 = kb.next();

} catch (Exception e) {

System.out.println("Error!!. Wrong input. Plese try agin..");

}

File dir = new File("DemoRepository\\"+userInput2);

dir.mkdir(); // Create a folder

System.out.println(" Seccessfil: Folder created seccessfully");

break;

}

}

public static void deleteFile() { // To delete a file

File dir = new File("DemoRepository"); // The directory path

File[] dirList = dir.listFiles(); // A list of the files in the directory

System.out.print("Enter file name: ");

Scanner kb = new Scanner(System.in);

String userInput="";

try {

userInput = kb.nextLine();

} catch (Exception e) {

System.out.println("Error!!. Wrong input. Plese try agin..");

}

boolean notFoundState = true; // stays 'true' in case Failed to delete the file

for(int i=0 ; i < dirList.length ; i++) {

if( dirList[i].getName().equals(userInput)) {

File myObj = new File("DemoRepository\\"+userInput);

if (myObj.delete()) {

System.out.println(" Succsesful: File deleted succsesfully");

notFoundState = false; // false to skip failure massage

break;

}

}

}

if(notFoundState) { // if not deleted, print a massage of failure

System.out.println(" Failed: file not found");

}

}

public static void showFiles(int choice) { // Choice '1' To List the files in the directory. Or '2' to Search for a file

File dir = new File("DemoRepository"); // The directory path

File[] dirList = dir.listFiles(); // A list of the files in the directory

if(choice == 1) { // List the files in the directory

for(int i=0 ; i < dirList.length ; i++) {

System.out.println(" file: "+ dirList[i].getName());

}

}

else if(choice == 2) { // Search for a file

System.out.print("Enter file name: ");

Scanner kb = new Scanner(System.in);

String userInput="";

try {

userInput = kb.nextLine();

} catch (Exception e) {

System.out.println("Error!!. Wrong input. Plese try agin..");

}

boolean notFoundState = true; // stays 'true' in case Failed to find

for(int i=0 ; i < dirList.length ; i++) {

if( dirList[i].getName().equals(userInput)) {

System.out.println(" Succsesful: File is found");

notFoundState = false; // false to skip failure massage

break;

}

}

if(notFoundState) { // if not found, print a massage of failure

System.out.println(" Failed: file not found");

}

}

}

public static void main(String[] args) {

new RVK();

}

}