***EJERCICIO FUNCIONES Nº 2***

Considere que es el tesorero de la junta de vecinos de su población. En conjunto con la directiva, se ha decidido abrir una cuenta vista en un banco, a la cual tendrán acceso un número X de personas (definidas por la directiva). La cuenta contempla solo dos tipos de movimientos bancarios, Giros y Depósitos de dinero. Solo podrán hacer giros aquellos que estén autorizados por la directiva, pero todos pueden hacer depósitos. Todos los movimientos (giros y depósitos) se realizan a través de Internet. Los accesos se controlan a través de un nombre de usuario y contraseña, los cuales deben ser creados al principio, cuando se abre la cuenta vista. El nombre de usuario es un número de tres dígitos, mientras que la contraseña es un número de 6 dígitos.

Construya un programa que gestione el movimiento de la cuenta bancaria, que: controle los accesos, valide el tipo de movimiento de acuerdo a los “permisos” de cada usuario y de acuerdo a los montos de dineros existentes en la cuenta, y registre todos los movimientos realizados.

Al finalizar, el programa deberá entregar un informe con todos los movimientos registrados, mostrando los siguientes datos: Nº operación, tipo de operación (giro o depósito), monto de la operación, saldo al final de la operación y usuario que realizó la operación.

Para esto debe utilizar al menos las siguientes funciones:

* Validar acceso: esta función debe validar que el nombre de usuario y contraseña sean los correctos para permitir el acceso.
* Mostrar informe: esta función muestra el informe final con la estructura solicitada
* Registrar operación: función que registra en el historial la nueva operación, ya sea giro o depósito, de acuerdo a la estructura solicitada.
* Validar operación: esta función debe validar que la operación esté permitida para un determinado usuario.
* Giro: función que realiza la operación de retiro de dinero.
* Depósito: función que realiza la operación de ingreso de dinero

Obs:

* Debe validar las entradas
* La primera operación no puede ser un Giro
* No puede sacar más dinero del que hay en la cuenta

***RESPUESTA***

package respev6m4;

import java.util.Scanner;

public class RespEv6M4 {

public static void main(String[] args) {

// TODO code application logic here

int numUs, opc, nomUs, opcOp;

System.out.println("Numero de Usuarios válidos");

numUs = leerDato();

int usuarios[][] = new int [numUs][3];

int registros[][] = new int [100][5];

crearUsuarios(usuarios,numUs);

do{

opc = 1;

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

System.out.println("Bienvenido al sistema Bancario");

System.out.println("Ingrese Usuario y Contraseña");

nomUs = validarIngreso(usuarios,numUs);

if (nomUs!=0){

do{

System.out.println("Bienvenido al sistema Bancario");

System.out.println("1.- Giro");

System.out.println("2.- Deposito");

System.out.println("3.- Salir");

System.out.println("Ingrese su opcion");

opcOp = leerDato();

switch (opcOp){

case 1: giro(registros,numUs,nomUs,usuarios);

break;

case 2: deposito(registros,nomUs);

break;

default: break;

}

}while(opcOp!=3);

System.out.println("Desea Ingresar con un nuevo Usuario? (1: si - 2 no)");

opc = leerDato();

}

}while((nomUs==0)||(opc==1));

mostrarRegistros(registros);

}

public static int leerDato(){

int dato;

Scanner leer = new Scanner(System.in);

System.out.println("Ingrese valor: ");

dato = leer.nextInt();

return dato;

}

public static void crearUsuarios(int [][] usuarios, int cantUs){

int i, nomUs, passUs, giro;

for(i=0;i<cantUs;i++){

do{

System.out.println("Nombre de Usuario "+(i+1));

nomUs = leerDato();

}while(nomUs<100 || nomUs>999);

do{

System.out.println("Contraseña "+(i+1));

passUs = leerDato();

}while(passUs<100000 || passUs>999999);

do{

System.out.println("Permite Giro (0: No - 1: Si) Usuario "+(i+1));

giro = leerDato();

}while(giro!=1 && giro!=0);

usuarios[i][0] = nomUs;

usuarios[i][1] = passUs;

usuarios[i][2] = giro;

}

}

public static int validarIngreso(int [][] usuarios, int cantUs){

int leeUs, leePass, fila;

System.out.println("Ingrese Nombre de usuario");

leeUs = leerDato();

fila = buscarUs(usuarios,cantUs, leeUs);

if (fila < cantUs){

System.out.println("Ingrese Password");

leePass = leerDato();

if (usuarios[fila][1]==leePass){

System.out.println("Acceso Permitido");

}

else{

System.out.println("Acceso Rechazado");

leeUs = 0;

}

}

else{

System.out.println("Usuario no existe");

leeUs = 0;

}

return leeUs;

}

public static int buscarUs(int [][] usuarios, int nUs, int us){

int i;

for(i=0;i<nUs;i++)

if (usuarios[i][0]==us)

return i;

return i;

}

public static void giro(int [][] registros, int cantUs, int idUs, int [][] usuarios){

int fila, montoGiro;

fila = buscarUs(usuarios,cantUs,idUs);

if (usuarios[fila][2]==1){

System.out.println("Giro Permitido");

System.out.println("Ingrese monto a girar");

montoGiro = leerDato();

registrarOp(registros,montoGiro,idUs,1);

}

else

System.out.println("Giro No Permitido - Usuario no autorizado");

}

public static void deposito(int [][] registros, int idUs){

int montoDep;

System.out.println("Ingrese monto a depositar");

montoDep = leerDato();

registrarOp(registros,montoDep,idUs,2);

}

public static void registrarOp(int [][] registros, int monto, int id, int tipoOp){

int i=0;

while(registros[i][0]!=0){

i++;

System.out.println("Valor de i "+i);

}

if(registros[i][0]==0){

registros[i][0]=i+1;

registros[i][1]=tipoOp;

registros[i][2]=monto;

if (tipoOp==1){

if (i==0)

System.out.println("La primera operación no puede ser Giro");

else{

if (monto>registros[i-1][3]){

System.out.println("Operación no permitida");

System.out.println("No hay suficiente dinero en la cuenta");

}

else{

registros[i][3]=registros[i-1][3]-monto;

System.out.println("Operación exitosa");

}

}

}

else{

if (i==0){

registros[i][3]=monto;

System.out.println("Operación exitosa");

}

else{

registros[i][3]=registros[i-1][3]+monto;

System.out.println("Operación exitosa");

}

}

registros[i][4]=id;

}

}

public static void mostrarRegistros(int [][] registros){

int i=0,j;

while(registros[i][0]!=0){

for(j=0;j<5;j++)

System.out.print(registros[i][j]+"\t");

System.out.println();

i++;

}

}

}