

Manual técnico

Muddap

Ana Clara Culeddu | Proyecto II | 5to 2da

# Código

//LIBRERIAS------------------------------------------------------------------------------------------------------------------------------------------------------

#include <iostream>

#include <stdlib.h>

#include <fstream>

#include <cstdio>

#include <string>

using namespace std;

//CLASES----------------------------------------------------------------------------------------------------------------------------------------------------------

class Usuarios{

private:

string nombre, contrasena;

public:

void setNombre(string nom);

void setContrasena(string con);

string getNombre();

string getContrasena();

Usuarios(string, string);

void crear(Usuarios user);

void registrar(Usuarios user);

};

class Cajas : public Usuarios{

private:

string nombre\_caja;

string tipo;

public:

void setNombre\_caja(string nom);

void setTipo(string ti);

string getNombre\_caja();

string getTipo();

Cajas(string, string, string, string);

void crear(Cajas caja, Usuarios user);

void modificar(Cajas caja, Usuarios user);

void borrar(Cajas caja, Usuarios user);

void mirar(Cajas caja, Usuarios user);

void cant\_cajas(Cajas caja, Usuarios user);

};

class Elementos: public Usuarios{

private:

string nombre\_elemento;

public:

void setNombreElemento(string nom);

string getNombreElemento();

Elementos(string, string, string);

void crear(Elementos elemento1, Cajas caja, Usuarios user);

void borrar(Elementos elemento1, Cajas caja, Usuarios user);

//bool verificarentero(string num1);

};

//CONSTRUCTOR SET GETTER - USUARIOS ---------------------------------------------------------------------------------------------------

Usuarios::Usuarios(string \_nombre, string \_contrasena){

nombre = \_nombre;

contrasena = \_contrasena;

};

void Usuarios::setNombre(string nom){

nombre = nom;

}

void Usuarios::setContrasena(string con){

contrasena = con;

}

string Usuarios::getNombre(){

return nombre;

}

string Usuarios::getContrasena(){

return contrasena;

}

//CONSTRUCTOR SET GETTER - CAJAS ---------------------------------------------------------------------------------------------------------

Cajas::Cajas(string \_nombre, string \_contrasena, string \_nombre\_caja, string \_tipo) : Usuarios(\_nombre, \_contrasena){

nombre\_caja = \_nombre\_caja;

tipo = \_tipo;

}

void Cajas::setNombre\_caja(string nom){

nombre\_caja = nom;

}

void Cajas::setTipo(string ti){

tipo = ti;

}

string Cajas::getNombre\_caja(){

return nombre\_caja;

}

string Cajas::getTipo(){

return tipo;

}

//CONSTRUCTOR SET GETTER - ELEMENTOS -----------------------------------------------------------------------------------------------------

Elementos::Elementos(string \_nombre, string \_contrasena, string \_nombre\_elemento) : Usuarios (\_nombre, \_contrasena){

nombre\_elemento = \_nombre\_elemento;

}

void Elementos::setNombreElemento(string nom){

nombre\_elemento = nom;

}

string Elementos::getNombreElemento(){

return nombre\_elemento;

}

//MENÚ ELEMENTOS --------------------------------------------------------------------------------------------------------------------------

void cambiarCaja(Cajas caja, Usuarios user){

system("cls");

Elementos elemento1 (user.getContrasena(), user.getNombre(), "");

bool volver = false;

string opcion;

while (volver == false){

cout<<"\n 1 - Agregar elementos \n 2 - Borrar elemento \n 3 - Volver atras \n Ingrese opcion: ";

cin>>opcion;

if (opcion=="1"){

elemento1.crear(elemento1, caja, user);

}

else if(opcion=="2"){

elemento1.borrar(elemento1, caja, user);

}

else if(opcion == "3"){

volver = true;

}

else{

cout << "Esa opción no existe" << endl;

}

};

}

//MENÚ CAJAS ---------------------------------------------------------------------------------------------------------------------------------

void menu(Usuarios user){

Cajas caja1 (user.getContrasena(), user.getNombre(),"", "");

string opcion, buscar;

bool salida = false;

ofstream archivo;

while (salida == false){

cout<<"\n 1 - Crear caja \n 2 - Borrar caja \n 3 - Modificar caja \n 4 - Mirar interior \n 5 - Ver todas las cajas \n 6 - Salir \n Ingrese opcion: ";

cin>>opcion;

if (opcion == "1"){

caja1.crear(caja1, user);

}

else if(opcion == "2"){

caja1.borrar(caja1, user);

}

else if(opcion == "3"){

caja1.modificar(caja1, user);

}

else if(opcion == "4"){

caja1.mirar(caja1, user);

}

else if(opcion == "5"){

caja1.cant\_cajas(caja1, user);

}

else if(opcion == "6"){

salida = true;

}

else{

cout<<"Esa opcion no existe.";

}

}

}

//MÉTODOS - USUARIOS -------------------------------------------------------------------------------------------------------------------------

bool verifica(string nom){

ifstream leer("usuarios.txt", ios::in);

string \_nom;

string \_con;

leer>>\_con;

while(!leer.eof()){

leer>>\_nom;

if(\_nom == nom){

leer.close();

return false;

}

}

leer.close();

return true;

}

bool verificar(string nom, string con){

ifstream leer("usuarios.txt", ios::in);

string \_nom;

string \_con;

while(!leer.eof()){

leer>>\_con;

leer>>\_nom;

if(\_nom == nom && \_con == con){

leer.close();

return false;

}

}

leer.close();

return true;

}

void Usuarios::crear(Usuarios user){

system("cls");

ofstream archivo1;

string nom, con;

cout << "Ingrese nombre de usuario: ";

cin >> nom;

cout << "Ingrese contrasena: ";

cin >> con;

archivo1.open("usuarios.txt", ios::app);

user.setContrasena(con);

user.setNombre(nom);

if (verifica(nom)){

archivo1 << user.getContrasena() + " " + user.getNombre() << endl;

cout << "¡Registro exitoso!" << endl;

}

else{

cout << "Ese nombre de usuario ya existe. Ingrese otro." << endl;

}

archivo1.close();

}

void Usuarios::registrar(Usuarios user){

system("cls");

ofstream archivo;

string nom, con;

cout << "Ingrese nombre de usuario: ";

cin >> nom;

cout << "Ingrese contrasena: ";

cin >> con;

if(verificar(nom, con)){

cout << "El usuario y la contraseña no coinciden" << endl;

}

else{

user.setContrasena(con);

user.setNombre(nom);

menu(user);

}

archivo.close();

}

//MÉTODOS - CAJAS ---------------------------------------------------------------------------------------------------------------------

void Cajas::crear(Cajas caja, Usuarios user){

system("cls");

string op\_crear, nombre, arch, c;

ofstream archivo;

ofstream ca;

cout<<"\n 1 - Caja para mudar \n 2 - Caja para donar \n Ingrese opcion: ";

cin>>op\_crear;

if (op\_crear == "1"){

caja.setTipo("mudar");

cout<<"Ingrese nombre de la caja: ";

cin>>nombre;

caja.setNombre\_caja(nombre);

arch = user.getNombre() + caja.getNombre\_caja() + caja.getTipo();

archivo.open(arch + ".txt", ios::app);

archivo.close();

c = "cajas" + user.getNombre();

ca.open(c + ".txt", ios::app);

ca << arch << endl;

ca.close();

}

else if(op\_crear == "2"){

caja.setTipo("donar");

cout<<"Ingrese nombre de la caja: ";

cin>>nombre;

caja.setNombre\_caja(nombre);

arch = user.getNombre() + caja.getNombre\_caja() + caja.getTipo();

archivo.open(arch + ".txt", ios::app);

archivo.close();

c = "cajas" + user.getNombre();

ca.open(c + ".txt", ios::app);

ca << arch << endl;

ca.close();

}

else{

cout<<"Esa opción no existe";

}

}

void Cajas::modificar(Cajas caja, Usuarios user){

system("cls");

string op\_crear, nombre, arch, c, nom;

bool encontrado = false;

ifstream archivo;

cout<<"\n 1 - Caja para mudar \n 2 - Caja para donar \n Ingrese opcion: ";

cin>>op\_crear;

if (op\_crear == "1"){

caja.setTipo("mudar");

cout<<"Ingrese nombre de la caja: ";

cin>>nombre;

caja.setNombre\_caja(nombre);

arch = user.getNombre() + caja.getNombre\_caja() + caja.getTipo();

c = "cajas" + user.getNombre() + ".txt";

archivo.open(c, ios::in);

if(archivo.is\_open()){

while(!archivo.eof()){

archivo >> nom;

if (nom == arch){

encontrado = true;

cambiarCaja(caja, user);

}

archivo >> nom;

}

}

archivo.close();

if (encontrado == false){

cout << "Caja no encontrada" << endl;

}

}

else if(op\_crear == "2"){

caja.setTipo("donar");

cout<<"Ingrese nombre de la caja: ";

cin>>nombre;

caja.setNombre\_caja(nombre);

arch = user.getNombre() + caja.getNombre\_caja() + caja.getTipo();

c = "cajas" + user.getNombre() + ".txt";

archivo.open(c, ios::in);

if(archivo.is\_open()){

while(!archivo.eof()){

archivo >> nom;

if (nom == arch){

encontrado = true;

cambiarCaja(caja, user);

}

archivo >> nom;

}

}

archivo.close();

if (encontrado == false){

cout << "Caja no encontrada" << endl;

}

}

else{

cout<<"Esa opción no existe";

}

}

void Cajas::borrar(Cajas caja, Usuarios user){

system("cls");

string op\_crear, nombre, arch, nom, c;

bool encontrado = false;

ifstream viejo;

ofstream nuevo;

cout<<"\n 1 - Caja para mudar \n 2 - Caja para donar \n Ingrese opcion: ";

cin>>op\_crear;

if (op\_crear == "1"){

caja.setTipo("mudar");

cout<<"Ingrese nombre de la caja: ";

cin>>nombre;

caja.setNombre\_caja(nombre);

arch = user.getNombre() + caja.getNombre\_caja() + caja.getTipo() + ".txt";

remove(arch.c\_str());

arch = user.getNombre() + caja.getNombre\_caja() + caja.getTipo();

c = "cajas" + user.getNombre() + ".txt";

viejo.open(c, ios::in);

nuevo.open("auxiliar.txt", ios::out);

if(viejo.is\_open()){

while(!viejo.eof()){

viejo >> nom;

if (nom == arch){

cout << "Registro borrado";

encontrado = true;

}

else{

nuevo << nom << endl;

}

viejo >> nom;

}

}

viejo.close();

nuevo.close();

remove(c.c\_str());

rename("auxiliar.txt", c.c\_str());

if (encontrado == false){

cout << "Registro no encontrado" << endl;

}

}

else if(op\_crear == "2"){

caja.setTipo("donar");

cout<<"Ingrese nombre de la caja: ";

cin>>nombre;

caja.setNombre\_caja(nombre);

arch = user.getNombre() + caja.getNombre\_caja() + caja.getTipo() + ".txt";

remove(arch.c\_str());

arch = user.getNombre() + caja.getNombre\_caja() + caja.getTipo();

c = "cajas" + user.getNombre() + ".txt";

viejo.open(c, ios::in);

nuevo.open("auxiliar.txt", ios::out);

if(viejo.is\_open()){

while(!viejo.eof()){

viejo >> nom;

if (nom == arch){

cout << "Registro borrado";

encontrado = true;

}

else{

nuevo << nom << endl;

}

viejo >> nom;

}

}

viejo.close();

nuevo.close();

remove(c.c\_str());

rename("auxiliar.txt", c.c\_str());

if (encontrado == false){

cout << "Registro no encontrado" << endl;

}

}

else{

cout<<"Esa opción no existe";

}

}

void Cajas::mirar(Cajas caja, Usuarios user){

system("cls");

string op\_crear, nombre, arch, si;

ifstream salida;

cout<<"\n 1 - Caja para mudar \n 2 - Caja para donar \n Ingrese opcion: ";

cin>>op\_crear;

if (op\_crear == "1"){

caja.setTipo("mudar");

cout<<"Ingrese nombre de la caja: ";

cin>>nombre;

caja.setNombre(nombre);

arch = user.getNombre() + caja.getNombre() + caja.getTipo() + ".txt";

salida.open(arch.c\_str(), ios::in);

}

else if(op\_crear == "2"){

caja.setTipo("donar");

cout<<"Ingrese nombre de la caja: ";

cin>>nombre;

caja.setNombre(nombre);

arch = user.getNombre() + caja.getNombre() + caja.getTipo() + ".txt";

salida.open(arch.c\_str(), ios::in);

}

else{

cout<<"Esa opción no existe";

}

cout << "-----------" << endl;

while(getline(salida, si)){

cout<<si<<endl;

}

cout << "-----------" << endl;

salida.close();

}

void Cajas::cant\_cajas(Cajas caja, Usuarios user){

system("cls");

string si, c;

ifstream salida;

c = "cajas" + user.getNombre();

salida.open(c + ".txt", ios::in);

cout << "-----------" << endl;

while(getline(salida, si)){

cout<<si<<endl;

}

cout << "-----------" << endl;

salida.close();

}

//MÉTODOS - ELEMENTOS -----------------------------------------------------------------------------------------------------------------------------------------

bool verificarentero(string num1)

{

try

{

int num2;

num2 = stoi(num1);

return true;

}

catch (...)

{

cout << "Error, ingrese un entero" << '\n';

return false;

}

};

void Elementos::crear(Elementos elemento1, Cajas caja, Usuarios user){

system("cls");

ofstream archivo;

string cant\_elementos, elemento, arch;

int numero;

arch = user.getNombre() + caja.getNombre\_caja() + caja.getTipo() + ".txt";

archivo.open(arch.c\_str(), ios::app);

cout<<"¿Cuantos elementos quiere ingresar?: ";

cin>>cant\_elementos;

verificarentero(cant\_elementos);

numero = stoi(cant\_elementos);

if (numero > 15){

cout << "No se puede agregar mas de 15 elementos a la vez";

}

else{

cout<<"Ingrese elemento: ";

for (int i=0; i<numero; i++){

cin>>elemento;

archivo<<elemento<<endl;

}

}

archivo.close();

}

void Elementos::borrar(Elementos elemento1, Cajas caja, Usuarios user){

system("cls");

string objeto, ele, arch;

bool encontrado = false;

ifstream viejo;

ofstream nuevo;

arch = user.getNombre() + caja.getNombre\_caja() + caja.getTipo() + ".txt";

viejo.open(arch, ios::in);

nuevo.open("auxiliar.txt", ios::out);

cout<<"Ingrese elemento a borrar: ";

cin>>objeto;

if(viejo.is\_open()){

while(!viejo.eof()){

viejo >> ele;

if (ele == objeto){

cout << "Registro borrado" << endl;

//encontrado = true;

}

else{

nuevo << ele << endl;

}

viejo >> ele;

}

}

viejo.close();

nuevo.close();

remove(arch.c\_str());

rename("auxiliar.txt", arch.c\_str());

/\*if (encontrado == false){

cout << "Registro no encontrado" << endl;

}\*/

}

//MAIN ------------------------------------------------------------------------------------------------------------------------------------------------------------

int main(){

Usuarios user ("", "");

string op\_entrada;

bool fin = false;

while (fin == false){

cout<<" \n ----------------------- \n ¡Bienvenidos a Mudapp! \n 1 - Registrarse \n 2 - Iniciar sesion \n 3 - Salir \n Ingrese opcion: ";

cin>>op\_entrada;

if (op\_entrada == "1"){

user.crear(user);

}

else if(op\_entrada == "2"){

user.registrar(user);

}

else if(op\_entrada == "3"){

cout << "¡Nos vemos pronto!";

fin = true;

}

else{

cout << "Esa opcion no existe. Ingrese de nuevo.";

}

}

system("pause");

return 0;

}