// ProyectoPedidosCarbon.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include "iostream"

using namespace std;

int TipoUsuarioActual;

int CodigoUsuarioActual;

struct regUsuario

{

int codigo;

char nombre[30];

char contrasenha[20];

int tipoUsuario;//1 Admin,2 Productor,3 Transportista,4 Cliente

char estado;

};

struct regCiudad

{

int codigo;

char nombre[30];

char estado;

};

struct regPlanta

{

int codigo;

char nombre[30];

int tipo\_producto;

int codigo\_productor;

int codigo\_ciudad;

char estado;

};

struct regRuta

{

int codigo;

int codigo\_ciudad\_A;

int codigo\_ciudad\_B;

char estado;

};

struct regTransportistaRuta

{

int codigo;

int codigo\_transportista;

int codigo\_ruta;

char estado;

};

struct regCompra

{

int codigo;

int codigo\_planta;

int codigo\_transportista;

int codigo\_cliente;

char estado;

};

int LeerEntero(char msj[])

{

int num;

do

{

cout << endl << msj;

cin >> num;

fflush(stdin);

cin.clear();

}while(num<=0);

return num;

}

void LeerCadena(char msj[],char cad[])

{

do{

cout << endl << msj;

gets(cad);

} while (strlen(cad)==0);

}

void solicitarUsuario(regUsuario \*reg)

{

reg->codigo = LeerEntero("Introducir Codigo:");

LeerCadena("Introducir Nombre:",reg->nombre);

LeerCadena("Introducir Contrasenha:",reg->contrasenha);

if(TipoUsuarioActual == 1)

{

do

{

reg->tipoUsuario = LeerEntero("Introducir tipo de usuario (1 Admin, 2 Productor,3 Transportista, 4 Cliente):");

}while(reg->tipoUsuario > 4);

}

else

{

do

{

reg->tipoUsuario = LeerEntero("Introducir tipo de usuario (2 Productor,3 Transportista, 4 Cliente):");

}while(reg->tipoUsuario > 4 || reg->tipoUsuario == 1);

}

reg->estado='N';

}

void mostrarUsuario(regUsuario reg, int nroReg)

{

cout<<endl<< nroReg<<".-"<< reg.nombre<<" "<<reg.tipoUsuario;

}

void adicionarAdministrador(char nomArch[15])

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch, "ab");

reg.codigo = 1;

strcpy(reg.nombre, "Administrador");

strcpy(reg.contrasenha, "1000");

reg.tipoUsuario = 1;

reg.estado = 'N';

fwrite(&reg, sizeof(reg),1,ptr);

fclose(ptr);

}

void adicionarUsuario(char nomArch[15])

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch, "ab");

solicitarUsuario(&reg);

fwrite(&reg, sizeof(reg),1,ptr);

fclose(ptr);

}

void listarUsuario(char nomArch[15])

{

FILE \*ptr;

regUsuario reg;

int cont=0;

ptr= fopen(nomArch,"rb");

if(ptr == NULL)

{

cout<< "EL archivo no existe...";

}

else

{

cout<<endl<<" \*\* LISTADO DE Usuario \*\*";

cout<<endl<<" Codigo Nombre Tipo";

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

fread(&reg, sizeof(reg),1,ptr);

while(!feof(ptr))

{

if(reg.estado=='N')

{

mostrarUsuario(reg,cont+1);

}

fread(&reg, sizeof(reg),1,ptr);//lee un registro del archivo

cont++;

}

fclose(ptr);

}

}

void buscarUsuario(char nomArch[15], int nroReg)

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch,"rb");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);//posiciona el ptr en un reg especifico

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

cout<<endl<< "Rrgistro no existe";

else

if(reg.estado=='N')

mostrarUsuario(reg,nroReg);

else

cout<<endl<< "Registro eliminado...";

fclose(ptr);

}

}

bool verificarContrasenha(char nomArch[15], int nroReg,char contrasenha[])

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch,"rb");

bool correcto = false;

if(ptr== NULL)

{

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);//posiciona el ptr en un reg especifico

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N')

{

if(strcmp(reg.contrasenha,contrasenha)==0)

{

cout<<endl<<"Bienvenido, "<<reg.nombre;

TipoUsuarioActual = reg.tipoUsuario;

CodigoUsuarioActual = reg.codigo;

correcto = true;

}

else

{

cout<<endl<<"contrasenha incorrecta";

}

}

else

{

cout<<endl<<"usuario eliminado";

}

}

fclose(ptr);

}

return correcto;

}

bool existeUsuario(char nomArch[15], int nroReg)

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N')

{

existe = true;

}

}

fclose(ptr);

}

return existe;

}

bool existeUsuarioProductor(char nomArch[15], int nroReg)

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N' && reg.tipoUsuario == 2)

{

existe = true;

}

}

fclose(ptr);

}

return existe;

}

bool existeUsuarioTransportista(char nomArch[15], int nroReg)

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N' && reg.tipoUsuario == 3)

{

existe = true;

}

}

fclose(ptr);

}

return existe;

}

bool existeUsuarioCliente(char nomArch[15], int nroReg)

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N' && reg.tipoUsuario == 4)

{

existe = true;

}

}

fclose(ptr);

}

return existe;

}

void modificarUsuario(char nomArch[15], int nroReg)

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Rrgistro no existe";

}

else

{

if(reg.estado=='N')

{

mostrarUsuario(reg,nroReg);

solicitarUsuario(&reg);

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

}

else

{

cout<<endl<< "Registro eliminado...";

}

fclose(ptr);

}

}

}

void eliminarUsuario(char nomArch[15], int nroReg)

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Registro no existe";

}

else

{

if(reg.estado=='N')

{

mostrarUsuario(reg,nroReg);

reg.estado= 'S';

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

cout<<endl<< "Registro eliminado exitosamente";

}

else

{

cout<<endl<< "Registro ya eliminado...";

}

fclose(ptr);

}

}

}

void solicitarCiudad(regCiudad \*reg)

{

reg->codigo = LeerEntero("Introducir Codigo:");

LeerCadena("Introducir Nombre:",reg->nombre);

reg->estado='N';

}

void mostrarCiudad(regCiudad reg, int nroReg)

{

cout<<endl<< nroReg<<".-"<< reg.nombre;

}

void adicionarCiudad(char nomArch[15])

{

FILE \*ptr;

regCiudad reg;

ptr= fopen(nomArch, "ab");

solicitarCiudad(&reg);

fwrite(&reg, sizeof(reg),1,ptr);

fclose(ptr);

}

void listarCiudad(char nomArch[15])

{

FILE \*ptr;

regCiudad reg;

int cont=0;

ptr= fopen(nomArch,"rb");

if(ptr == NULL)

{

cout<< "EL archivo no existe...";

}

else

{

cout<<endl<<" \*\* LISTADO DE Ciudad \*\*";

cout<<endl<<" Codigo Nombre";

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

fread(&reg, sizeof(reg),1,ptr);

while(!feof(ptr))

{

if(reg.estado=='N')

{

mostrarCiudad(reg,cont+1);

}

fread(&reg, sizeof(reg),1,ptr);

cont++;

}

fclose(ptr);

}

}

void buscarCiudad(char nomArch[15], int nroReg)

{

FILE \*ptr;

regCiudad reg;

ptr= fopen(nomArch,"rb");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

cout<<endl<< "Rrgistro no existe";

else

if(reg.estado=='N')

mostrarCiudad(reg,nroReg);

else

cout<<endl<< "Registro eliminado...";

fclose(ptr);

}

}

bool existeCiudad(char nomArch[15], int nroReg)

{

FILE \*ptr;

regCiudad reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N')

{

existe = true;

}

}

fclose(ptr);

}

return existe;

}

void modificarCiudad(char nomArch[15], int nroReg)

{

FILE \*ptr;

regCiudad reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Rrgistro no existe";

}

else

{

if(reg.estado=='N')

{

mostrarCiudad(reg,nroReg);

solicitarCiudad(&reg);

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

}

else

{

cout<<endl<< "Registro eliminado...";

}

fclose(ptr);

}

}

}

void eliminarCiudad(char nomArch[15], int nroReg)

{

FILE \*ptr;

regCiudad reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Registro no existe";

}

else

{

if(reg.estado=='N')

{

mostrarCiudad(reg,nroReg);

reg.estado= 'S';

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

cout<<endl<< "Registro eliminado exitosamente";

}

else

{

cout<<endl<< "Registro ya eliminado...";

}

fclose(ptr);

}

}

}

void solicitarPlanta(regPlanta \*reg)

{

reg->codigo = LeerEntero("Introducir Codigo:");

LeerCadena("Introducir Nombre:",reg->nombre);

reg->tipo\_producto = LeerEntero("Introducir tipo producto (1 Carbon negro, 2 Carbon gris):");

if(TipoUsuarioActual == 1)

{

do{

reg->codigo\_productor = LeerEntero("Introducir Codigo de usuario productor:");

}while(!existeUsuarioProductor("Usuario.dat",reg->codigo\_productor ));

}

if(TipoUsuarioActual == 2)

{

reg->codigo\_productor = CodigoUsuarioActual;

}

do{

reg->codigo\_ciudad = LeerEntero("Introducir Codigo de ciudad:");

}while(!existeCiudad("Ciudad.dat",reg->codigo\_ciudad ));

reg->estado='N';

}

void mostrarPlanta(regPlanta reg, int nroReg)

{

cout<<endl<< nroReg<<".-"<< reg.nombre << " " << reg.codigo\_ciudad<< " " << reg.codigo\_productor;

if(reg.tipo\_producto == 1)

{

cout<<" carbon negro";

}

else

{

cout<<" carbon gris";

}

}

void adicionarPlanta(char nomArch[15])

{

FILE \*ptr;

regPlanta reg;

ptr= fopen(nomArch, "ab");

solicitarPlanta(&reg);

fwrite(&reg, sizeof(reg),1,ptr);

fclose(ptr);

}

void listarPlanta(char nomArch[15])

{

FILE \*ptr;

regPlanta reg;

int cont=0;

ptr= fopen(nomArch,"rb");

if(ptr == NULL)

{

cout<< "EL archivo no existe...";

}

else

{

cout<<endl<<" \*\* LISTADO DE Planta \*\*";

cout<<endl<<" Codigo Nombre Ciudad Productor tipo producto";

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

fread(&reg, sizeof(reg),1,ptr);

while(!feof(ptr))

{

if(reg.estado=='N')

{

mostrarPlanta(reg,cont+1);

}

fread(&reg, sizeof(reg),1,ptr);

cont++;

}

fclose(ptr);

}

}

void buscarPlanta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regPlanta reg;

ptr= fopen(nomArch,"rb");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

cout<<endl<< "Rrgistro no existe";

else

if(reg.estado=='N')

mostrarPlanta(reg,nroReg);

else

cout<<endl<< "Registro eliminado...";

fclose(ptr);

}

}

bool existePlanta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regPlanta reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N')

{

existe = true;

}

}

fclose(ptr);

}

return existe;

}

void modificarPlanta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regPlanta reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Rrgistro no existe";

}

else

{

if(reg.estado=='N')

{

if(TipoUsuarioActual == 1 || (TipoUsuarioActual == 2 && CodigoUsuarioActual == reg.codigo\_productor))

{

mostrarPlanta(reg,nroReg);

solicitarPlanta(&reg);

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

}

else

{

cout<<endl<<"Usted no tiene permiso para modificar la planta asignada a otro productor";

}

}

else

{

cout<<endl<< "Registro eliminado...";

}

fclose(ptr);

}

}

}

void eliminarPlanta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regPlanta reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Registro no existe";

}

else

{

if(reg.estado=='N')

{

if(TipoUsuarioActual == 1 || (TipoUsuarioActual == 2 && CodigoUsuarioActual == reg.codigo\_productor))

{

mostrarPlanta(reg,nroReg);

reg.estado= 'S';

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

cout<<endl<< "Registro eliminado exitosamente";

}

else

{

cout<<endl<<"Usted no tiene permiso para eliminar la planta asignada a otro productor";

}

}

else

{

cout<<endl<< "Registro ya eliminado...";

}

fclose(ptr);

}

}

}

void solicitarRuta(regRuta \*reg)

{

reg->codigo = LeerEntero("Introducir Codigo:");

do{

reg->codigo\_ciudad\_A = LeerEntero("Introducir Codigo de ciudad A:");

}while(!existeCiudad("Ciudad.dat",reg->codigo\_ciudad\_A ));

do{

reg->codigo\_ciudad\_B = LeerEntero("Introducir Codigo de ciudad B:");

}while(!existeCiudad("Ciudad.dat",reg->codigo\_ciudad\_B ));

reg->estado='N';

}

void mostrarRuta(regRuta reg, int nroReg)

{

cout<<endl<< nroReg<<".-"<< " " << reg.codigo\_ciudad\_A << " " << reg.codigo\_ciudad\_B;

}

void adicionarRuta(char nomArch[15])

{

FILE \*ptr;

regRuta reg;

ptr= fopen(nomArch, "ab");

solicitarRuta(&reg);

fwrite(&reg, sizeof(reg),1,ptr);

fclose(ptr);

}

void listarRuta(char nomArch[15])

{

FILE \*ptr;

regRuta reg;

int cont=0;

ptr= fopen(nomArch,"rb");

if(ptr == NULL)

{

cout<< "EL archivo no existe...";

}

else

{

cout<<endl<<" \*\* LISTADO DE Ruta \*\*";

cout<<endl<<" Codigo Nombre Ciudad A Ciudad B";

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

fread(&reg, sizeof(reg),1,ptr);

while(!feof(ptr))

{

if(reg.estado=='N')

{

mostrarRuta(reg,cont+1);

}

fread(&reg, sizeof(reg),1,ptr);

cont++;

}

fclose(ptr);

}

}

void buscarRuta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regRuta reg;

ptr= fopen(nomArch,"rb");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

cout<<endl<< "Rrgistro no existe";

else

if(reg.estado=='N')

mostrarRuta(reg,nroReg);

else

cout<<endl<< "Registro eliminado...";

fclose(ptr);

}

}

bool existeRuta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regRuta reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N')

{

existe = true;

}

}

fclose(ptr);

}

return existe;

}

bool existeRutaTransportista(char nomArch[15], int nroReg)

{

FILE \*ptr;

regRuta reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N')

{

existe = true;

}

}

fclose(ptr);

}

return existe;

}

void modificarRuta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regRuta reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Rrgistro no existe";

}

else

{

if(reg.estado=='N')

{

mostrarRuta(reg,nroReg);

solicitarRuta(&reg);

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

}

else

{

cout<<endl<< "Registro eliminado...";

}

fclose(ptr);

}

}

}

void eliminarRuta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regRuta reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Registro no existe";

}

else

{

if(reg.estado=='N')

{

mostrarRuta(reg,nroReg);

reg.estado= 'S';

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

cout<<endl<< "Registro eliminado exitosamente";

}

else

{

cout<<endl<< "Registro ya eliminado...";

}

fclose(ptr);

}

}

}

void MenuUsuario()

{

int op;

int busc;

do

{

cout<<endl;

cout<<endl<<"Menu Usuario";

cout<<endl<<"1. Adicionar Usuario";

cout<<endl<<"2. Buscar Usuario";

cout<<endl<<"3. Modificar Usuario";

cout<<endl<<"4. Listar Usuario";

cout<<endl<<"5. Eliminar Usuario";

cout<<endl<<"0. Salir";

cout<<endl<<"opcion:";

cin>>op;

switch(op)

{

case 1:

adicionarUsuario("Usuario.dat");

break;

case 2:

busc = LeerEntero("Usuario buscada:");

buscarUsuario("Usuario.dat",busc);

break;

case 3:

busc = LeerEntero("Usuario para modificar:");

modificarUsuario("Usuario.dat",busc);

break;

case 4:

listarUsuario("Usuario.dat");

break;

case 5:

busc = LeerEntero("Usuario para eliminar:");

eliminarUsuario("Usuario.dat",busc);

break;

default:

break;

}

}while(op!=0);

}

void MenuCiudad()

{

int op;

int busc;

do

{

cout<<endl;

cout<<endl<<"Menu Ciudad";

cout<<endl<<"1. Adicionar Ciudad";

cout<<endl<<"2. Buscar Ciudad";

cout<<endl<<"3. Modificar Ciudad";

cout<<endl<<"4. Listar Ciudad";

cout<<endl<<"5. Eliminar Ciudad";

cout<<endl<<"0. Salir";

cout<<endl<<"opcion:";

cin>>op;

switch(op)

{

case 1:

adicionarCiudad("Ciudad.dat");

break;

case 2:

busc = LeerEntero("Ciudad buscada:");

buscarCiudad("Ciudad.dat",busc);

break;

case 3:

busc = LeerEntero("Ciudad para modificar:");

modificarCiudad("Ciudad.dat",busc);

break;

case 4:

listarCiudad("Ciudad.dat");

break;

case 5:

busc = LeerEntero("Ciudad para elliminar:");

eliminarCiudad("Ciudad.dat",busc);

break;

default:

break;

}

}while(op!=0);

}

void MenuPlanta()

{

int op;

int busc;

do

{

cout<<endl;

cout<<endl<<"Menu Planta";

cout<<endl<<"1. Adicionar Planta";

cout<<endl<<"2. Buscar Planta";

cout<<endl<<"3. Modificar Planta";

cout<<endl<<"4. Listar Planta";

cout<<endl<<"5. Eliminar Planta";

cout<<endl<<"0. Salir";

cout<<endl<<"opcion:";

cin>>op;

switch(op)

{

case 1:

adicionarPlanta("Planta.dat");

break;

case 2:

busc = LeerEntero("Planta buscada:");

buscarPlanta("Planta.dat",busc);

break;

case 3:

busc = LeerEntero("Planta para modificar:");

modificarPlanta("Planta.dat",busc);

break;

case 4:

listarPlanta("Planta.dat");

break;

case 5:

busc = LeerEntero("Planta para eliminar:");

eliminarPlanta("Planta.dat",busc);

break;

default:

break;

}

}while(op!=0);

}

void MenuRuta()

{

int op;

int busc;

do

{

cout<<endl;

cout<<endl<<"Menu Ruta";

cout<<endl<<"1. Adicionar Ruta";

cout<<endl<<"2. Buscar Ruta";

cout<<endl<<"3. Modificar Ruta";

cout<<endl<<"4. Listar Ruta";

cout<<endl<<"5. Eliminar Ruta";

cout<<endl<<"0. Salir";

cout<<endl<<"opcion:";

cin>>op;

switch(op)

{

case 1:

adicionarRuta("Ruta.dat");

break;

case 2:

busc = LeerEntero("Ruta buscada:");

buscarRuta("Ruta.dat",busc);

break;

case 3:

busc = LeerEntero("Ruta para modificar:");

modificarRuta("Ruta.dat",busc);

break;

case 4:

listarRuta("Ruta.dat");

break;

case 5:

busc = LeerEntero("Ruta para eliminar:");

eliminarRuta("Ruta.dat",busc);

break;

default:

break;

}

}while(op!=0);

}

void solicitarTransportistaRuta(regTransportistaRuta \*reg)

{

reg->codigo = LeerEntero("Introducir Codigo:");

if(TipoUsuarioActual == 1)

{

do{

reg->codigo\_transportista = LeerEntero("Introducir Codigo transportista:");

}while(!existeUsuarioTransportista("Usuario.dat",reg->codigo\_transportista ));

}

if(TipoUsuarioActual == 3)

{

reg->codigo\_transportista = CodigoUsuarioActual;

}

do{

reg->codigo\_ruta = LeerEntero("Introducir Codigo ruta:");

}while(!existeRuta("Ruta.dat",reg->codigo\_ruta ));

reg->estado='N';

}

void mostrarTransportistaRuta(regTransportistaRuta reg, int nroReg)

{

cout<<endl<< nroReg<<".-"<< reg.codigo\_ruta << " " << reg.codigo\_transportista;

}

void adicionarTransportistaRuta(char nomArch[15])

{

FILE \*ptr;

regTransportistaRuta reg;

ptr= fopen(nomArch, "ab");

solicitarTransportistaRuta(&reg);

fwrite(&reg, sizeof(reg),1,ptr);

fclose(ptr);

}

void listarTransportistaRuta(char nomArch[15])

{

FILE \*ptr;

regTransportistaRuta reg;

int cont=0;

ptr= fopen(nomArch,"rb");

if(ptr == NULL)

{

cout<< "EL archivo no existe...";

}

else

{

cout<<endl<<" \*\* LISTADO DE Transportista-Ruta \*\*";

cout<<endl<<" Codigo Trasnportista Ruta";

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

fread(&reg, sizeof(reg),1,ptr);

while(!feof(ptr))

{

if(reg.estado=='N')

{

mostrarTransportistaRuta(reg,cont+1);

}

fread(&reg, sizeof(reg),1,ptr);

cont++;

}

fclose(ptr);

}

}

void buscarTransportistaRuta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regTransportistaRuta reg;

ptr= fopen(nomArch,"rb");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

cout<<endl<< "Rrgistro no existe";

else

if(reg.estado=='N')

mostrarTransportistaRuta(reg,nroReg);

else

cout<<endl<< "Registro eliminado...";

fclose(ptr);

}

}

bool existeTransportistaRuta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regTransportistaRuta reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N')

{

existe = true;

}

}

fclose(ptr);

}

return existe;

}

void modificarTransportistaRuta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regTransportistaRuta reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Rrgistro no existe";

}

else

{

if(reg.estado=='N')

{

if(TipoUsuarioActual == 1 || (TipoUsuarioActual == 3 && CodigoUsuarioActual == reg.codigo\_transportista))

{

mostrarTransportistaRuta(reg,nroReg);

solicitarTransportistaRuta(&reg);

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

}

else

{

cout<<endl<<"Usted no tiene permiso para modificar la ruta asignada a otro transportista";

}

}

else

{

cout<<endl<< "Registro eliminado...";

}

fclose(ptr);

}

}

}

void eliminarTransportistaRuta(char nomArch[15], int nroReg)

{

FILE \*ptr;

regTransportistaRuta reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Registro no existe";

}

else

{

if(reg.estado=='N')

{

if(TipoUsuarioActual == 1 || (TipoUsuarioActual == 3 && CodigoUsuarioActual == reg.codigo\_transportista))

{

mostrarTransportistaRuta(reg,nroReg);

reg.estado= 'S';

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

cout<<endl<< "Registro eliminado exitosamente";

}

else

{

cout<<endl<<"Usted no tiene permiso para eliminar la ruta asignada a otro transportista";

}

}

else

{

cout<<endl<< "Registro ya eliminado...";

}

fclose(ptr);

}

}

}

void MenuTransportistaRuta()

{

int op;

int busc;

do

{

cout<<endl;

cout<<endl<<"Menu TransportistaRuta";

cout<<endl<<"1. Adicionar Transportista-Ruta";

cout<<endl<<"2. Buscar Transportista-Ruta";

cout<<endl<<"3. Modificar Transportista-Ruta";

cout<<endl<<"4. Listar Transportista-Ruta";

cout<<endl<<"5. Eliminar Transportista-Ruta";

cout<<endl<<"0. Salir";

cout<<endl<<"opcion:";

cin>>op;

switch(op)

{

case 1:

adicionarTransportistaRuta("TransportistaRuta.dat");

break;

case 2:

busc = LeerEntero("Transportista-Ruta buscada:");

buscarTransportistaRuta("TransportistaRuta.dat",busc);

break;

case 3:

busc = LeerEntero("Transportista-Ruta para modificar:");

modificarTransportistaRuta("TransportistaRuta.dat",busc);

break;

case 4:

listarTransportistaRuta("TransportistaRuta.dat");

break;

case 5:

busc = LeerEntero("Transportista-Ruta para modificar:");

eliminarTransportistaRuta("TransportistaRuta.dat",busc);

break;

default:

break;

}

}while(op!=0);

}

bool ExisteTransportistaRuta(char nomArch[],int transportista,int ruta)

{

FILE \*ptr;

regTransportistaRuta reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr == NULL)

{

cout<< "EL archivo no existe...";

}

else

{

fread(&reg, sizeof(reg),1,ptr);

while(!feof(ptr))

{

if(reg.estado=='N')

{

if(reg.codigo\_ruta == ruta && reg.codigo\_transportista == transportista)

{

existe = true;

}

}

fread(&reg, sizeof(reg),1,ptr);

}

fclose(ptr);

}

return existe;

}

int EncontrarRuta(char nomArch[],int ciudadA,int ciudadB)

{

FILE \*ptr;

regRuta reg;

ptr= fopen(nomArch,"rb");

int encontrada = 0;

if(ptr == NULL)

{

cout<< "EL archivo no existe...";

}

else

{

fread(&reg, sizeof(reg),1,ptr);

while(!feof(ptr))

{

if(reg.estado=='N')

{

if(reg.codigo\_ciudad\_A == ciudadA && reg.codigo\_ciudad\_B == ciudadB)

{

encontrada = reg.codigo;

}

if(reg.codigo\_ciudad\_A == ciudadB && reg.codigo\_ciudad\_B == ciudadA)

{

encontrada = reg.codigo;

}

}

fread(&reg, sizeof(reg),1,ptr);

}

fclose(ptr);

}

return encontrada;

}

int EncontrarCiudadPlanta(char nomArch[],int planta)

{

FILE \*ptr;

regPlanta reg;

ptr= fopen(nomArch,"rb");

int ciudad = 0;

if(ptr == NULL)

{

cout<< "EL archivo no existe...";

}

else

{

fread(&reg, sizeof(reg),1,ptr);

while(!feof(ptr))

{

if(reg.estado=='N')

{

if(reg.codigo == planta)

{

ciudad = reg.codigo\_ciudad;

}

}

fread(&reg, sizeof(reg),1,ptr);

}

fclose(ptr);

}

return ciudad;

}

void MostrarTransportistasPosibles(char nomArch[],int ruta)

{

FILE \*ptr;

regUsuario reg;

ptr= fopen(nomArch,"rb");

int cont = 1;

if(ptr == NULL)

{

cout<< "EL archivo no existe...";

}

else

{

fread(&reg, sizeof(reg),1,ptr);

while(!feof(ptr))

{

if(reg.estado=='N')

{

if(ExisteTransportistaRuta("TransportistaRuta.dat",reg.codigo,ruta))

{

mostrarUsuario(reg,cont);

}

}

cont++;

fread(&reg, sizeof(reg),1,ptr);

}

fclose(ptr);

}

}

void solicitarCompra(regCompra \*reg)

{

reg->codigo = LeerEntero("Introducir Codigo:");

do{

reg->codigo\_planta = LeerEntero("Introducir Codigo planta:");

}while(!existePlanta("Planta.dat",reg->codigo\_planta ));

int ciudadA = EncontrarCiudadPlanta("Planta.dat",reg->codigo\_planta);

int ciudadB;

do{

ciudadB = LeerEntero("Introducir Ciudad a la que se desea entregar el pedido:");

}while(!existeCiudad("Ciudad.dat",ciudadB ));

int ruta = EncontrarRuta("Ruta.dat",ciudadA,ciudadB);

MostrarTransportistasPosibles("Usuario.dat",ruta);

do{

reg->codigo\_transportista = LeerEntero("Introducir Transportista de la lista (0 para salir):");

if(reg->codigo\_transportista == 0)

{

return;

}

}while(!ExisteTransportistaRuta("TransportistaRuta.dat",reg->codigo\_transportista,ruta));

if(TipoUsuarioActual == 1)

{

do{

reg->codigo\_cliente = LeerEntero("Introducir Codigo cliente:");

}while(!existeUsuarioTransportista("Usuario.dat",reg->codigo\_cliente ));

}

if(TipoUsuarioActual == 4)

{

reg->codigo\_cliente = CodigoUsuarioActual;

}

reg->estado='N';

}

void mostrarCompra(regCompra reg, int nroReg)

{

cout<<endl<< nroReg<<".-"<< reg.codigo\_planta << " " << reg.codigo\_transportista << " " << reg.codigo\_cliente;

}

void adicionarCompra(char nomArch[15])

{

FILE \*ptr;

regCompra reg;

ptr= fopen(nomArch, "ab");

solicitarCompra(&reg);

if(reg.codigo\_transportista > 0)

{

fwrite(&reg, sizeof(reg),1,ptr);

}

fclose(ptr);

}

void listarCompra(char nomArch[15])

{

FILE \*ptr;

regCompra reg;

int cont=0;

ptr= fopen(nomArch,"rb");

if(ptr == NULL)

{

cout<< "EL archivo no existe...";

}

else

{

cout<<endl<<" \*\* LISTADO DE Compra \*\*";

cout<<endl<<" Codigo Transportista Cliente";

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

fread(&reg, sizeof(reg),1,ptr);

while(!feof(ptr))

{

if(reg.estado=='N')

{

mostrarCompra(reg,cont+1);

}

fread(&reg, sizeof(reg),1,ptr);

cont++;

}

fclose(ptr);

}

}

void buscarCompra(char nomArch[15], int nroReg)

{

FILE \*ptr;

regCompra reg;

ptr= fopen(nomArch,"rb");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

cout<<endl<< "Rrgistro no existe";

else

if(reg.estado=='N')

mostrarCompra(reg,nroReg);

else

cout<<endl<< "Registro eliminado...";

fclose(ptr);

}

}

bool existeCompra(char nomArch[15], int nroReg)

{

FILE \*ptr;

regCompra reg;

ptr= fopen(nomArch,"rb");

bool existe = false;

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(!feof(ptr))

{

if(reg.estado=='N')

{

existe = true;

}

}

fclose(ptr);

}

return existe;

}

void modificarCompra(char nomArch[15], int nroReg)

{

FILE \*ptr;

regCompra reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Rrgistro no existe";

}

else

{

if(reg.estado=='N')

{

if(TipoUsuarioActual == 1 || (TipoUsuarioActual == 4 && CodigoUsuarioActual == reg.codigo\_cliente))

{

mostrarCompra(reg,nroReg);

solicitarCompra(&reg);

if(reg.codigo\_transportista > 0)

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

}

}

else

{

cout<<endl<<"Usted no tiene permiso para modificar la compra asignada a otro cliente";

}

}

else

{

cout<<endl<< "Registro eliminado...";

}

fclose(ptr);

}

}

}

void eliminarCompra(char nomArch[15], int nroReg)

{

FILE \*ptr;

regCompra reg;

ptr= fopen(nomArch, "r+b");

if(ptr== NULL)

{

cout<< "El archivo no existe...";

}

else

{

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fread(&reg, sizeof(reg),1,ptr);

if(feof(ptr))

{

cout<<endl<< "Registro no existe";

}

else

{

if(reg.estado=='N')

{

if(TipoUsuarioActual == 1 || (TipoUsuarioActual == 4 && CodigoUsuarioActual == reg.codigo\_cliente))

{

mostrarCompra(reg,nroReg);

reg.estado= 'S';

fseek(ptr, (nroReg-1)\*sizeof(reg),SEEK\_SET);

fwrite(&reg, sizeof(reg),1,ptr);

cout<<endl<< "Registro eliminado exitosamente";

}

else

{

cout<<endl<<"Usted no tiene permiso para eliminar la compra asignada a otro cliente";

}

}

else

{

cout<<endl<< "Registro ya eliminado...";

}

fclose(ptr);

}

}

}

void MenuCompra()

{

int op;

int busc;

do

{

cout<<endl<<"Menu Compra";

cout<<endl<<"1. Adicionar Compra";

cout<<endl<<"2. Buscar Compra";

cout<<endl<<"3. Modificar Compra";

cout<<endl<<"4. Listar Compra";

cout<<endl<<"5. Eliminar Compra";

cout<<endl<<"0. Salir";

cout<<endl<<"opcion:";

cin>>op;

switch(op)

{

case 1:

adicionarCompra("Compra.dat");

break;

case 2:

busc = LeerEntero("Compra buscada:");

buscarCompra("Compra.dat",busc);

break;

case 3:

busc = LeerEntero("Compra para modificar:");

modificarCompra("Compra.dat",busc);

break;

case 4:

listarCompra("Compra.dat");

break;

case 5:

busc = LeerEntero("Compra para modificar:");

eliminarCompra("Compra.dat",busc);

break;

default:

break;

}

}while(op!=0);

}

void MenuPrincipal()

{

int op;

int busc;

do

{

cout<<endl;

cout<<endl<<"Menu Principal";

if(TipoUsuarioActual == 1 || TipoUsuarioActual == 2)

{

cout<<endl<<"1. Administrar Plantas";

}

if(TipoUsuarioActual == 1 || TipoUsuarioActual == 3)

{

cout<<endl<<"2. Administrar Rutas";

cout<<endl<<"3. Administrar Rutas asignadas a transportistas";

}

if(TipoUsuarioActual == 1 || TipoUsuarioActual == 4)

{

cout<<endl<<"4. Administrar Compras";

}

if(TipoUsuarioActual == 1)

{

cout<<endl<<"5. Administrar Ciudades";

cout<<endl<<"6. Administrar Usuarios";

}

cout<<endl<<"0. Salir";

cout<<endl<<"opcion:";

cin>>op;

switch(op)

{

case 1:

if(TipoUsuarioActual == 1 || TipoUsuarioActual == 2)

{

MenuPlanta();

}

break;

case 2:

if(TipoUsuarioActual == 1 || TipoUsuarioActual == 3)

{

MenuRuta();

}

break;

case 3:

if(TipoUsuarioActual == 1 || TipoUsuarioActual == 3)

{

MenuTransportistaRuta();

}

break;

case 4:

if(TipoUsuarioActual == 1 || TipoUsuarioActual == 4)

{

MenuCompra();

}

break;

case 5:

if(TipoUsuarioActual == 1)

{

MenuCiudad();

}

break;

case 6:

if(TipoUsuarioActual == 1)

{

MenuUsuario();

}

break;

default:

break;

}

}while(op!=0);

}

int main()

{

int op;

int usuario;

char contrasenha[20];

TipoUsuarioActual = 0;

if(!existeUsuario("Usuario.dat",1))

{

adicionarAdministrador("Usuario.dat");

}

do

{

cout<<endl;

cout<<endl<<"---Proyecto Carbones---"<<endl;

cout<<endl<<"1.- Ingresar";

cout<<endl<<"2.- Nuevo";

cout<<endl<<"0. Salir";

cout<<endl<<"opcion:";

cin>>op;

switch(op)

{

case 1:

usuario = LeerEntero("Ingresar codigo de usuario");

LeerCadena("Ingresar contrasenha:",contrasenha);

system("cls");

if(verificarContrasenha("Usuario.dat",usuario,contrasenha))

{

MenuPrincipal();

TipoUsuarioActual = 0;

}

break;

case 2:

adicionarUsuario("Usuario.dat");

break;

default:

break;

}

}while(op!=0);

return 0;

}