#include <stdio.h>

#include <conio.h>

typedef struct nodo{

int codigo;

char nombre[20];

int precio;

struct nodo\* siguiente;

} nodo;

nodo\* primero = NULL;

nodo\* ultimo = NULL;

void insertarProducto();

void buscarProducto();

void modificarProducto();

void eliminarProducto();

void desplegarLista();

int main(){

int opcionMenu = 0;

do{

printf("\n1.- Insertar.");

printf("\n2. Buscar.");

printf("\n3. Modificar.");

printf("\n4. Eliminar.");

printf("\n5. Desplegar.");

printf("\n6. Salir.");

printf("\n|------------------|------------------|");

printf("\n\n Escoja una Opcion: ");

scanf("%d", &opcionMenu);

switch(opcionMenu){

case 1:

printf("\n\n Insertar producto en la lista \n\n");

insertarProducto();

break;

case 2:

printf("\n\n Buscar producto en la lista \n\n");

buscarProducto();

break;

case 3:

printf("\n\n Modificar producto en la lista \n\n");

modificarProducto();

break;

case 4:

printf("\n\n Eliminar producto en la lista \n\n");

eliminarProducto();

break;

case 5:

printf("\n\n Desplegrar lista de productos \n\n");

desplegarLista();

break;

case 6:

printf("\n\n Programa finalizado...");

break;

default:

printf("\n\n Opcion No Valida \n\n");

}

}while(opcionMenu != 6);

return 0;

}

void insertarProducto(){

nodo\* nuevo = (nodo\*) malloc(sizeof(nodo));

printf(" Ingrese el codigo: ");

scanf("%d", &nuevo->codigo);

printf(" Ingrese el nombre: ");

scanf("%s", &nuevo->nombre);

printf(" Ingrese el precio: ");

scanf("%i", &nuevo->precio);

if(primero==NULL){

primero = nuevo;

primero->siguiente = NULL;

ultimo = nuevo;

}else{

ultimo->siguiente = nuevo;

nuevo->siguiente = NULL;

ultimo = nuevo;

}

printf("\n Producto ingresado con Exito\n\n");

}

void buscarProducto(){

nodo\* actual = (nodo\*) malloc(sizeof(nodo));

actual = primero;

int nodoBuscado = 0, encontrado = 0;

printf(" Ingrese el dato del nodo a Buscar: ");

scanf("%d", &nodoBuscado);

if(primero!=NULL){

while(actual != NULL && encontrado != 1){

if(actual->codigo == nodoBuscado){

printf("\n El codigo: ( %d ) Encontrado\n\n", nodoBuscado);

encontrado = 1;

}

actual = actual->siguiente;

}

if(encontrado == 0){

printf("\n Codigo no encontrado\n\n");

}

}else{

printf("\n La lista se encuentra vacia\n\n");

}

}

void modificarProducto(){

nodo\* actual = (nodo\*) malloc(sizeof(nodo));

actual = primero;

int nodoBuscado = 0, encontrado = 0;

printf(" Ingrese el codigo del producto que desea modificar: ");

scanf("%d", &nodoBuscado);

if(primero!=NULL){

while(actual != NULL && encontrado != 1){

if(actual->codigo == nodoBuscado){

printf("\n El nodo con el dato ( %d ) Encontrado", nodoBuscado);

printf("\n Ingrese el nuevo codigo para este Producto: ");

scanf("%d", &actual->codigo);

printf("\n Ingrese el nuevo nombre para este Producto: ");

scanf("%d", &actual->nombre);

printf("\n Ingrese el nuevo precio para este Producto: ");

scanf("%d", &actual->precio);

printf("\n Nodo modificado con exito\n\n");

encontrado = 1;

}

actual = actual->siguiente;

}

if(encontrado == 0){

printf("\n Nodo no encontrado\n\n");

}

}else{

printf("\n La lista se encuentra vacia\n\n");

}

}

void eliminarProducto(){

nodo\* actual = (nodo\*) malloc(sizeof(nodo));

actual = primero;

nodo\* anterior = (nodo\*) malloc(sizeof(nodo));

anterior = NULL;

int nodoBuscado = 0, encontrado = 0;

printf(" Ingrese el codigo a Buscar para Eliminar: ");

scanf("%d", &nodoBuscado);

if(primero!=NULL){

while(actual != NULL && encontrado != 1){

if(actual->codigo == nodoBuscado){

if(actual == primero){

primero = primero->siguiente;

}else if(actual == ultimo){

anterior->siguiente = NULL;

ultimo = anterior;

}else{

anterior->siguiente = actual->siguiente;

}

printf("\n Producto de codigo eliminado con exito");

encontrado = 1;

}

anterior = actual;

actual = actual->siguiente;

}

if(encontrado == 0){

printf("\n Codigo no encontrado\n\n");

}else{

free(anterior);

}

}else{

printf("\n La lista se encuentra vacia\n\n");

}

}

void desplegarLista(){

nodo\* actual = (nodo\*) malloc(sizeof(nodo));

actual = primero;

if(primero!=NULL){

while(actual != NULL){

printf("%d %s %i", actual->codigo, actual->nombre, actual->precio);

actual = actual->siguiente;

}

}else{

printf("\n La lista se encuentra vacia\n\n");

}

}

void Ventas(){

printf("No se han registrado ventas aun.");

}