

# HITO 2 DEL 1º TRIMESTRE DE Programación

Rodrigo García Ortiz

20 - 11 – 2024

# Índice

# Fase 1

El primer paso es crear la base de datos en la que se guardaran los datos del programa, creando tablas para los clientes, los productos y los pedidos.

Luego pasamos al código. Empezamos por crear la conexión a la base de datos.

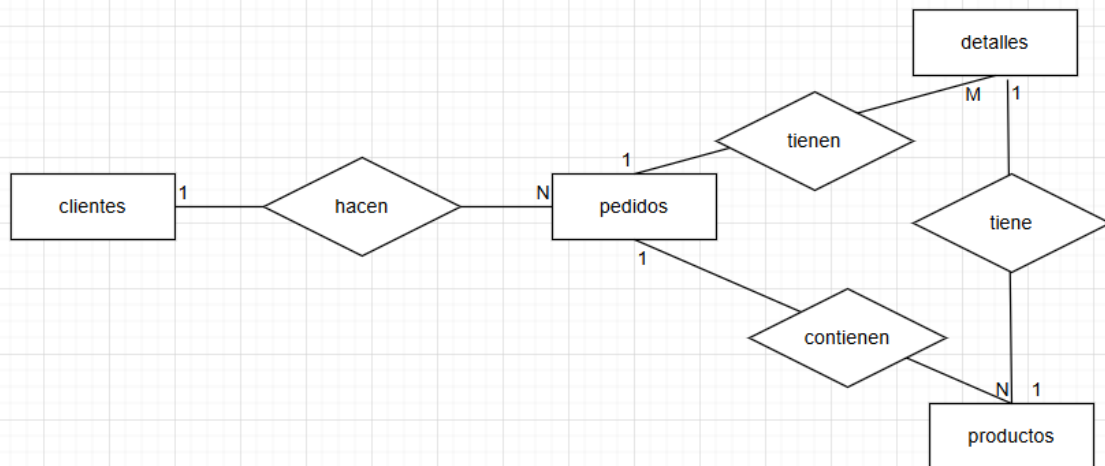
Después definimos las funciones: para registrar clientes, para mostrar los clientes registrados, para buscar un cliente en concreto, para mostrar los productos, para elegir los productos, para hacer un pedido y para guardar los detalles del pedido.

Después creamos un menú y definimos una función para escoger la opción deseada.

Por último, creamos una función que nos permita ver el menú y escoger opciones todas las veces que queramos.

- Importamos la conexión a la base de datos
- Definimos la función conectar para conectarnos a la base de datos
- Importamos la función a otro archivo que usaremos para las funciones del programa
- Definimos la función que registra clientes usando un insert into en la tabla clientes
- Definimos la función que muestra los clientes usando un select from la tabla clientes
- Definimos la función que permite buscar un cliente concreto usando un select from clientes
- Definimos la función que muestra los productos usando un select from productos
- Definimos la función que permite elegir productos usando un select from productos
- Definimos la función que permite hacer pedidos usando un insert into pedidos
- Definimos la función que muestra los detalles de un pedido usando select from pedidos inner join clientes inner join detalles
- Exportamos las funciones a otro archivo en el crearemos el menú
- Definimos la función que crea el menú usando print()
- Definimos la función que permite escoger opciones:
  - Una opción nos pide los datos para registrar un cliente
  - Una opción muestra a los clientes y nos pide el id del cliente que queremos buscar y nos lo muestra
  - Una opción nos muestra los productos y nos permite hacer un pedido usando un bucle while para que introduzcamos todos los productos deseados al pedido
  - Una opción nos muestra los datos del pedido que busquemos

- Una opción nos permite salir del programa
- Exportamos las funciones de menú a otro archivo
- Definimos una función que usa un bucle while para que hagamos todas las consultas que queramos



## Fase 2

bdd.py

```
import mysql.connector

def conectar(): #Conecta con la base de datos
    conexion = mysql.connector.connect(
        host="localhost",    # Cambia según tu configuración
        user="root",         # Cambia según tu configuración
        password="curso",    # Cambia según tu configuración
        database="gestion_pedidos"
    )
    return conexion
```

funciones.py

```
from bdd import conectar

def crear_cliente(nombre, telefono, direccion): #Registra un cliente
    conexion = conectar()
    cursor = conexion.cursor()
    consulta = "INSERT INTO clientes (nombre, telefono, direccion) VALUES (%s, %s, %s)"
    cursor.execute(consulta, (nombre, telefono, direccion))
    conexion.commit()
    cursor.close()
    conexion.close()

def leer_clientes(): #Muestra los clientes registrados
    conexion = conectar()
    cursor = conexion.cursor()
    cursor.execute("select * from clientes")
    clientes = cursor.fetchall()
    cursor.close()
    conexion.close()
    return clientes

def buscar_cliente(busqueda): # Muestra un cliente concreto
    conexion = conectar()
    cursor = conexion.cursor()
    consulta = """select nombre, telefono, direccion from clientes where id_cliente = %s"""
    cursor.execute(consulta, (busqueda,))
    resultado = cursor.fetchall()
    conexion.commit()
    cursor.close()
```

```

        conexion.close()
        return resultado

def mostrar_productos(): # Muestra los productos
    conexion = conectar()
    cursor = conexion.cursor()
    cursor.execute("select * from productos")
    productos = cursor.fetchall()
    conexion.commit()
    cursor.close()
    conexion.close()
    return productos

def comprar(producto): #Devuelve el precio del producto escogido
    conexion = conectar()
    cursor = conexion.cursor()
    consulta = "select precio_producto from productos where id_producto =
%s"
    cursor.execute(consulta, (producto,))
    precio = cursor.fetchall()
    precio_producto = float(precio[0][0])
    conexion.commit()
    cursor.close()
    conexion.close()
    return precio_producto

def crear_pedido(cliente, precio, numero_objetos): #Registra un pedido
    conexion = conectar()
    cursor = conexion.cursor()
    consulta = "INSERT INTO pedidos (cliente, precio, numero_objetos)
VALUES (%s, %s, %s)"
    cursor.execute(consulta, (cliente, precio, numero_objetos))
    id_pedido = cursor.lastrowid
    conexion.commit()
    cursor.close()
    conexion.close()
    return id_pedido

def seleccionar_pedido(id_pedido): #Muestra los datos de un pedido, su
cliente y sus detalles
    conexion = conectar()
    cursor = conexion.cursor()
    consulta = "select pedidos.*, clientes.* from pedidos inner join
clientes on id_cliente = cliente where id_pedido = %s"
    cursor.execute(consulta, (id_pedido,))
    pedido = cursor.fetchall()
    conexion.commit()
    cursor.close()
    conexion.close()

```

```
return pedido
```

menus.py

```
from funciones import crear_cliente, leer_clientes, buscar_cliente,
mostrar_productos, comprar, crear_pedido, seleccionar_pedido

def mostrar_menu(): #Muestra el menú
    print("\n=== Menú de opciones ===")
    print("1. Registrar clientes")
    print("2. Ver y buscar clientes")
    print("3. Realizar compra")
    print("4. Seguimiento de compra")
    print("5. Salir")

def ejecutar_opcion(opcion):#Realiza la opción asignada
    match opcion:
        case 1: #Registra un cliente con los datos introducidos
            nombre = input("Ingrese el nombre del nuevo cliente: ")
            telefono = int(input("Ingrese el teléfono del nuevo cliente:
"))
            direccion = input("Ingrese la dirección del nuevo cliente: ")
            crear_cliente(nombre, telefono, direccion)
            print("Cliente registrado exitosamente.")

        case 2: #Muestra los clientes
            clientes = leer_clientes()
            print("\nListado de Clientes:")
            for id_cliente, nombre, telefono, direccion in clientes:
                print(f"ID: {id_cliente}, Nombre: {nombre}, Teléfono:
{telefono} Dirección: {direccion}")
            busqueda = (input("Introduzca el ID del cliente que quiere
buscar: "))
            print(buscar_cliente(busqueda))

        case 3: #Registra un pedido asignado al cliente escogido con los
productos escogidos
            id_cliente = int(input("Indique el cliente que realizará la
compra: ")) #Escoge el cliente
            productos = mostrar_productos() #Selecciona los datos del
producto
            print("\nLista de Productos:")
            for id_producto, nombre_producto, precio_producto in
productos: #Muestra los productos
                print(f"ID: {id_producto} Nombre: {nombre_producto}
Precio: {precio_producto}€")
            precio_total = 0
            numero_objetos = 0
```

```

        while True:
            producto = int(input("Introduce el ID del producto que
quieres añadir al pedido: ")) #Introduce un producto al pedido
            if producto == 0:
                break
            else:
                precio_total = precio_total + comprar(producto)
                numero_objetos = numero_objetos + 1
            print(f"El precio total del pedido es: {precio_total}€")
            crear_pedido(id_cliente, precio_total, numero_objetos)
#Registra un pedido
            print(f"El número del pedido es: {crear_pedido(id_cliente,
precio_total, numero_objetos)}")

        case 4: #Muestra el número del pedido creado
            pedido = int(input("Introduce el ID del pedido deseado: "))
            seleccionar_pedido(pedido)

        case 5: #Cierra la aplicación
            return False

        case _: #Introduces un número sin opción asignada
            print("Opción inválida")

```

main.py

```

import menus

def main():
    continuar = True
    while continuar:
        menus.mostrar_menu() #Muestra el menú
        try:
            opcion = int(input("Seleccione una opción: ")) #Elige una
opcion
            menus.ejecutar_opcion(opcion) #Ejecuta la opción escogida

        except ValueError:
            print("Error: Debe ingresar un número válido.")

if __name__ == "__main__":
    main()

```



## Webgrafía

*Get the id after INSERT into MySQL database using Python.* (2020, diciembre 10). GeeksforGeeks. <https://www.geeksforgeeks.org/get-the-id-after-insert-into-mysql-database-using-python/>

*MySQL INNER JOIN keyword.* (s/f). W3schools.com. Recuperado el 21 de noviembre de 2024, de [https://www.w3schools.com/mysql/mysql\\_join\\_inner.asp](https://www.w3schools.com/mysql/mysql_join_inner.asp)

<https://github.com/Rodrigo-Garcia-Ortiz/Programacion.git>