

## ACTIVIDAD ELECTIVA II

La Biblioteca Municipal de San José era un lugar emblemático del centro histórico de la ciudad. Fundada en 1902, albergaba miles de libros entre novelas clásicas, poesía, teatro y estudios académicos, atrayendo a estudiantes, intelectuales y amantes de la lectura por generaciones.

Sin embargo, doña Elvira, la fiel bibliotecaria que llevaba 39 años organizando todas las colecciones solo con un enorme fichero de cartón amarillento, decidió jubilarse. Esto dejó un vacío inmenso en la biblioteca, ya que doña Elvira era la única que entendía dónde estaba cada libro, entre las altas y polvorientas estanterías de madera con libros apilados en desorden.

La alcaldesa de San José, preocupada por esta situación, reunió a su equipo determinada a modernizar la biblioteca. Así contactaron a un grupo de ingenieros de sistemas graduados en la Universidad Tecnológica para contratar el desarrollo e implementación de un sistema computarizado de gestión bibliotecaria.

Los ingenieros visitaron la biblioteca y se sorprendieron ante la monumental tarea de catalogar los más de 60.000 libros existentes. Sin embargo, motivados por el reto, se dispusieron a trabajar arduamente durante 6 meses, ingresando cuidadosamente el título, autor, y estado de cada ejemplar en una base de datos que programaron utilizando Mysql.

Finalmente, desarrollaron una aplicación usando Python. La alcaldesa quedó muy complacida con el sistema, que fue rápidamente adoptado por los bibliotecarios y por todo el público asiduo de la biblioteca, especialmente los estudiantes.

Este proyecto representó para los ingenieros un reto motivador que les permitió aplicar los conocimientos de programación, bases de datos y gestión de proyectos obtenidos durante su carrera. Además, revivió a esta histórica biblioteca de San José, atrayendo incluso a nuevas generaciones de lectores. Cada vez que los ingenieros visitan la biblioteca y ven el impacto positivo de su sistema no pueden evitar recordar con orgullo aquel formidable reto que superaron hace algunos años.

**Forma de entrega:** Enviar el código fuente en un archivo .zip a [chenao@coruniamericana.edu.co](mailto:chenao@coruniamericana.edu.co)

En base al anterior caso de estudio desarrollar un sistema de biblioteca que cumpla con las características anteriormente descritas.

#### **conexion.py:**

Es un módulo Python que contiene una función para conectarse a una base de datos MySQL.

Explica cómo realizar la conexión, manejar errores y devolver la conexión para usarla después.

#### **db.sql:**

Contiene instrucciones SQL para crear una tabla llamada "libro" en la base de datos.

Define la estructura de la tabla con columnas para id, título, autor y estado.

#### **libro.py:**

Importa la función de conexión del módulo conexion.py

Define funciones para registrar, mostrar, buscar, modificar y eliminar registros en la tabla "libro".

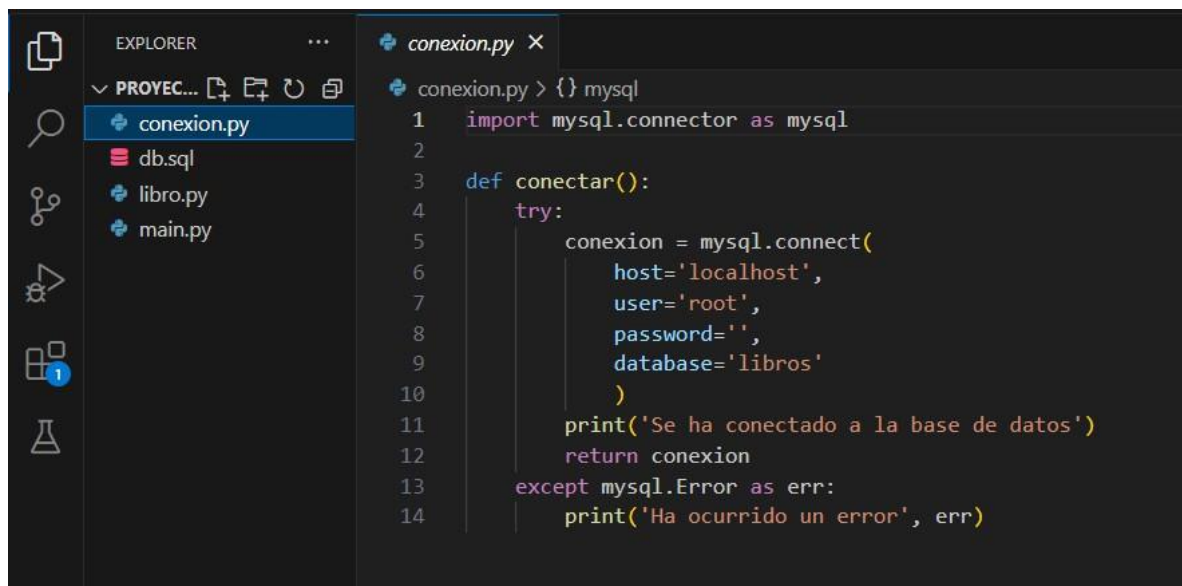
#### **main.py:**

Programa principal que importa módulos y llama a funciones.

Contiene un menú con opciones para las operaciones CRUD de la tabla libro.

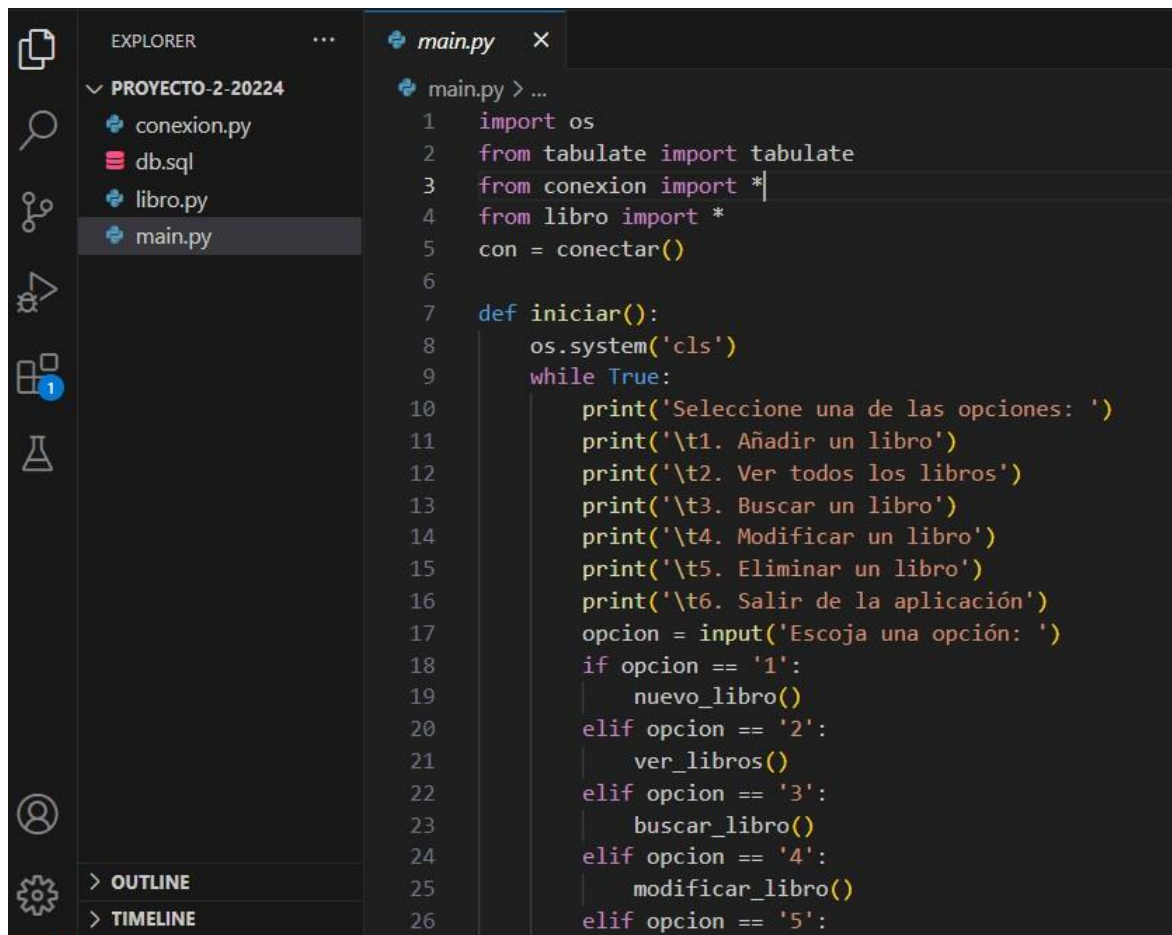
Usa tabulate para mostrar los datos en forma de tabla.

Comenzaremos creando conexión.py....

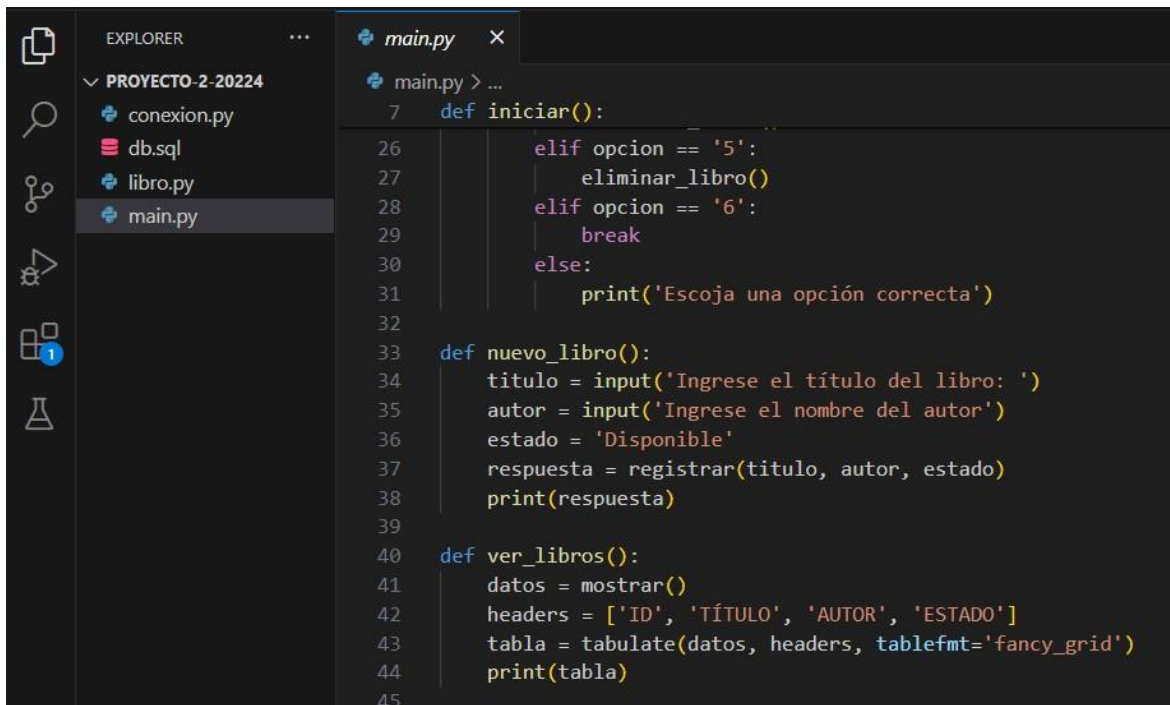


```
conexion.py > {} mysql
1 import mysql.connector as mysql
2
3 def conectar():
4     try:
5         conexion = mysql.connect(
6             host='localhost',
7             user='root',
8             password='',
9             database='libros'
10        )
11        print('Se ha conectado a la base de datos')
12        return conexion
13    except mysql.Error as err:
14        print('Ha ocurrido un error', err)
```

Luego el main.py

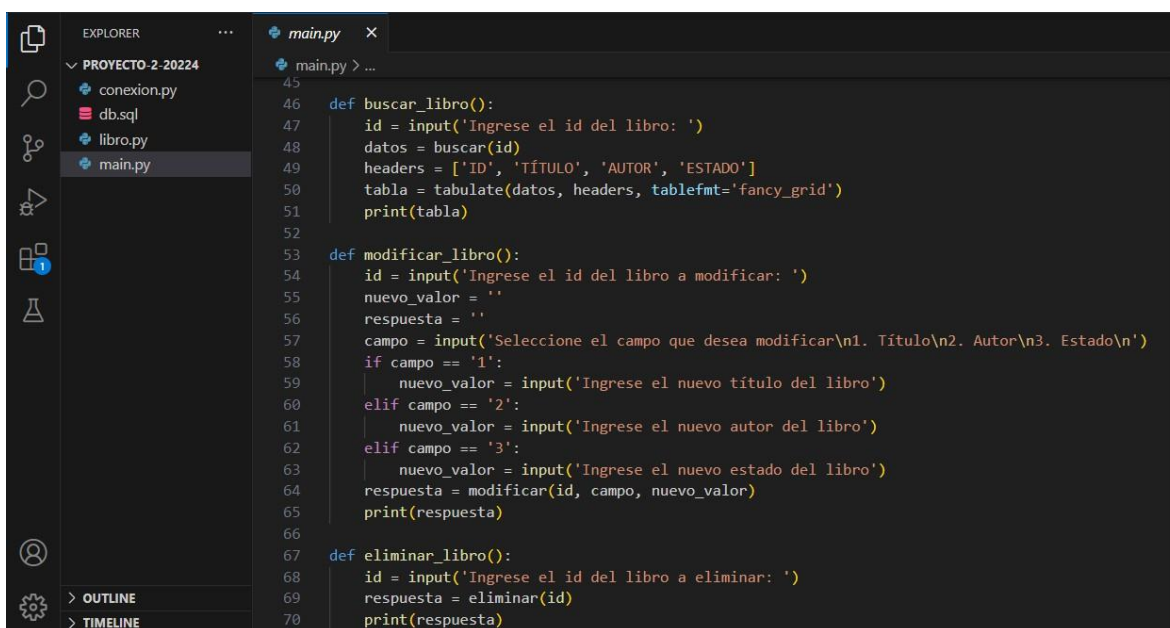


```
1 import os
2 from tabulate import tabulate
3 from conexion import *
4 from libro import *
5 con = conectar()
6
7 def iniciar():
8     os.system('cls')
9     while True:
10         print('Seleccione una de las opciones: ')
11         print('\t1. Añadir un libro')
12         print('\t2. Ver todos los libros')
13         print('\t3. Buscar un libro')
14         print('\t4. Modificar un libro')
15         print('\t5. Eliminar un libro')
16         print('\t6. Salir de la aplicación')
17         opcion = input('Escoja una opción: ')
18         if opcion == '1':
19             nuevo_libro()
20         elif opcion == '2':
21             ver_libros()
22         elif opcion == '3':
23             buscar_libro()
24         elif opcion == '4':
25             modificar_libro()
26         elif opcion == '5':
```



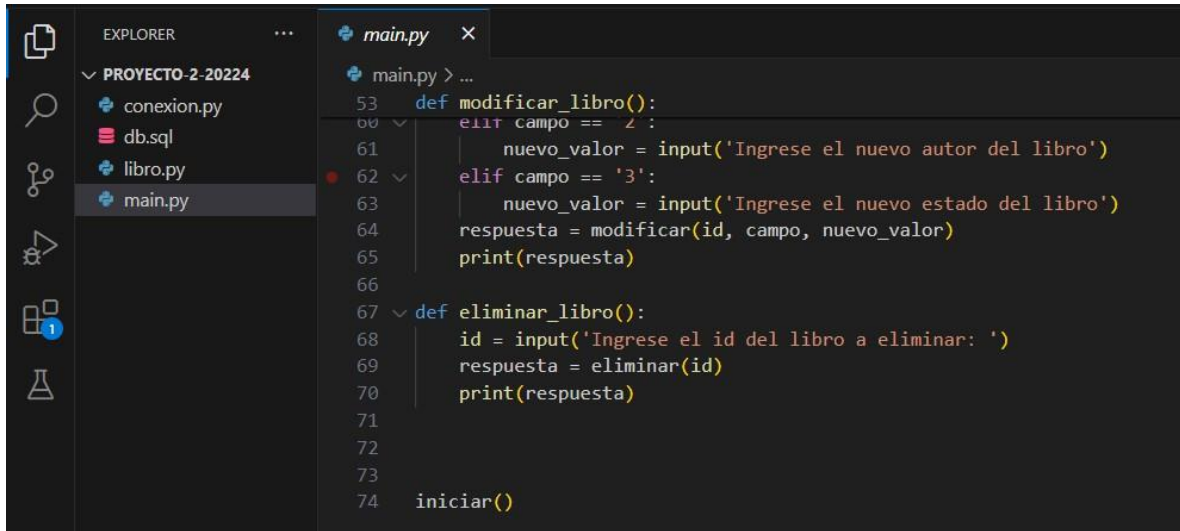
The screenshot shows the Visual Studio Code editor with a project named 'PROYECTO-2-20224'. The Explorer sidebar on the left lists files: 'conexion.py', 'db.sql', 'libro.py', and 'main.py'. The 'main.py' file is open in the editor, showing the following code:

```
main.py > ...
7  def iniciar():
26      elif opcion == '5':
27          eliminar_libro()
28      elif opcion == '6':
29          break
30      else:
31          print('Escoja una opción correcta')
32
33  def nuevo_libro():
34      titulo = input('Ingrese el título del libro: ')
35      autor = input('Ingrese el nombre del autor')
36      estado = 'Disponible'
37      respuesta = registrar(titulo, autor, estado)
38      print(respuesta)
39
40  def ver_libros():
41      datos = mostrar()
42      headers = ['ID', 'TÍTULO', 'AUTOR', 'ESTADO']
43      tabla = tabulate(datos, headers, tablefmt='fancy_grid')
44      print(tabla)
45
```



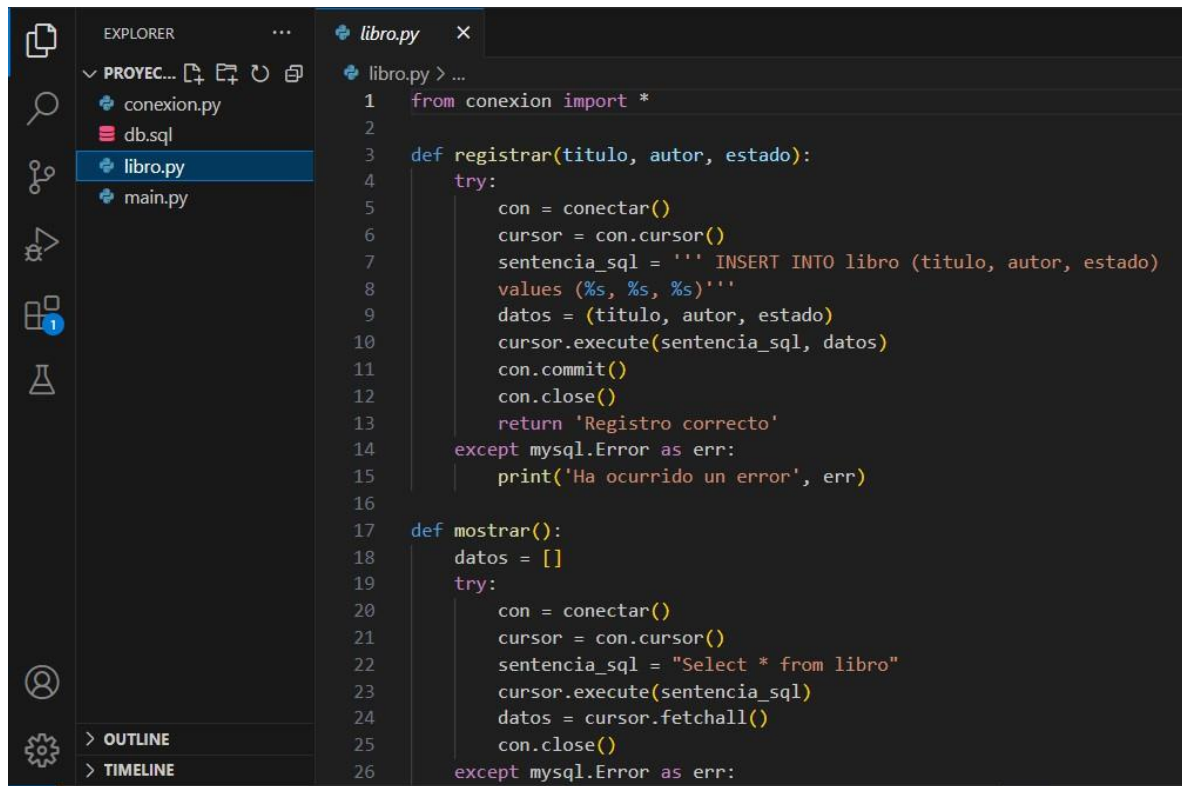
The screenshot shows the Visual Studio Code editor with the same project and file list. The 'main.py' file is open, showing the continuation of the code from the previous screenshot:

```
45
46  def buscar_libro():
47      id = input('Ingrese el id del libro: ')
48      datos = buscar(id)
49      headers = ['ID', 'TÍTULO', 'AUTOR', 'ESTADO']
50      tabla = tabulate(datos, headers, tablefmt='fancy_grid')
51      print(tabla)
52
53  def modificar_libro():
54      id = input('Ingrese el id del libro a modificar: ')
55      nuevo_valor = ''
56      respuesta = ''
57      campo = input('Seleccione el campo que desea modificar\n1. Título\n2. Autor\n3. Estado\n')
58      if campo == '1':
59          nuevo_valor = input('Ingrese el nuevo título del libro')
60      elif campo == '2':
61          nuevo_valor = input('Ingrese el nuevo autor del libro')
62      elif campo == '3':
63          nuevo_valor = input('Ingrese el nuevo estado del libro')
64      respuesta = modificar(id, campo, nuevo_valor)
65      print(respuesta)
66
67  def eliminar_libro():
68      id = input('Ingrese el id del libro a eliminar: ')
69      respuesta = eliminar(id)
70      print(respuesta)
```

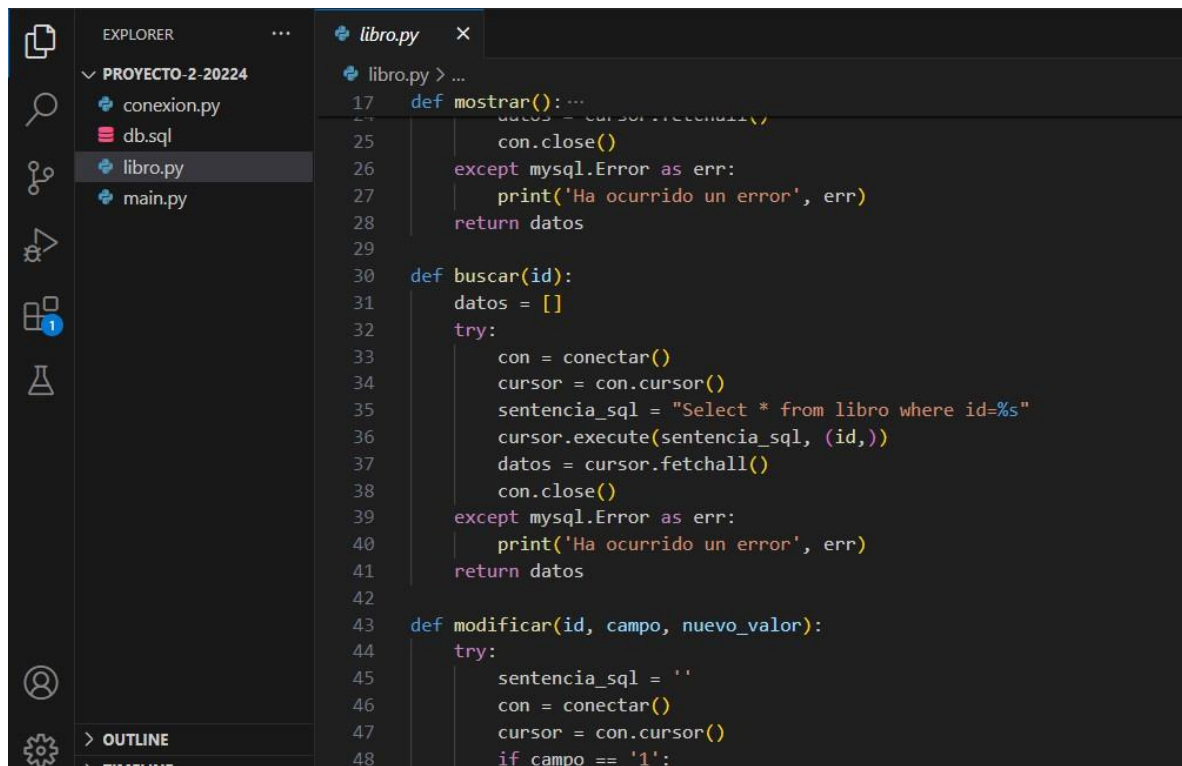


```
53 def modificar_libro():
54     elif campo == '2':
55         nuevo_valor = input('Ingrese el nuevo autor del libro')
56     elif campo == '3':
57         nuevo_valor = input('Ingrese el nuevo estado del libro')
58     respuesta = modificar(id, campo, nuevo_valor)
59     print(respuesta)
60
61
62
63
64
65
66
67 def eliminar_libro():
68     id = input('Ingrese el id del libro a eliminar: ')
69     respuesta = eliminar(id)
70     print(respuesta)
71
72
73
74 iniciar()
```

Luego libro.py

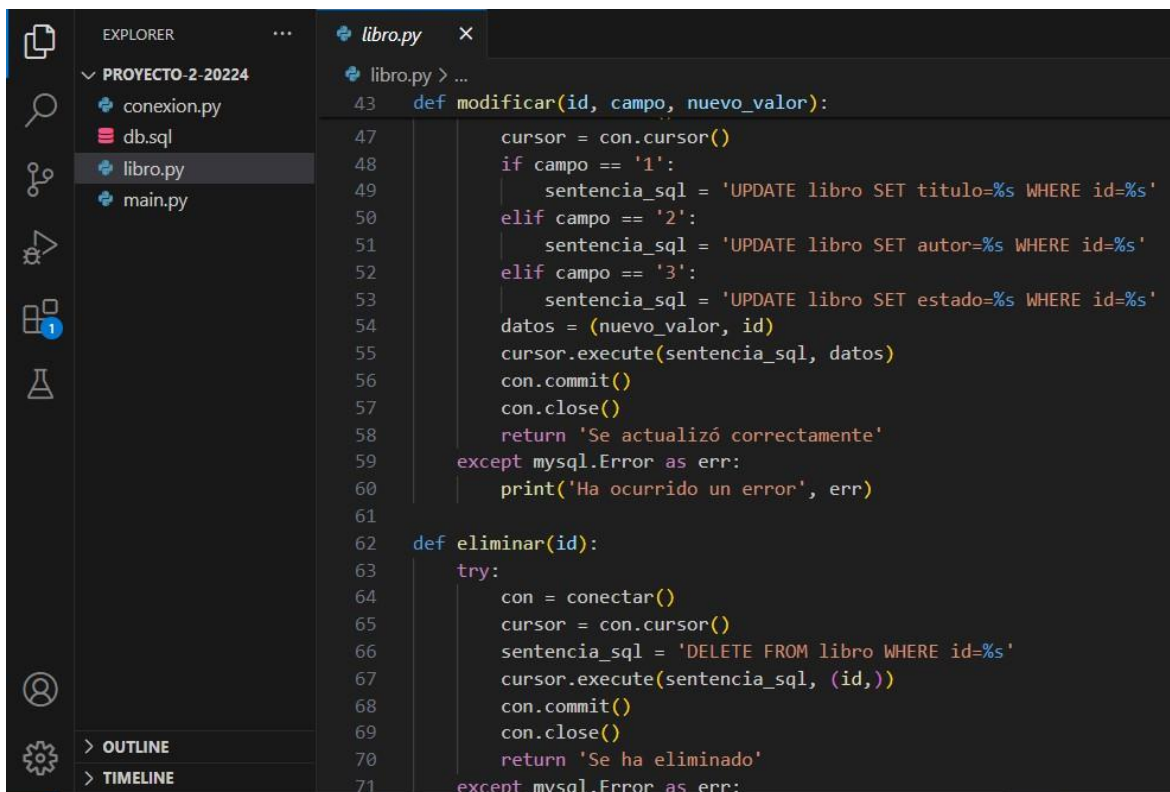


```
1 from conexion import *
2
3 def registrar(titulo, autor, estado):
4     try:
5         con = conectar()
6         cursor = con.cursor()
7         sentencia_sql = ''' INSERT INTO libro (titulo, autor, estado)
8         values (%s, %s, %s)'''
9         datos = (titulo, autor, estado)
10        cursor.execute(sentencia_sql, datos)
11        con.commit()
12        con.close()
13        return 'Registro correcto'
14    except mysql.Error as err:
15        print('Ha ocurrido un error', err)
16
17 def mostrar():
18     datos = []
19     try:
20         con = conectar()
21         cursor = con.cursor()
22         sentencia_sql = "Select * from libro"
23         cursor.execute(sentencia_sql)
24         datos = cursor.fetchall()
25         con.close()
26     except mysql.Error as err:
```

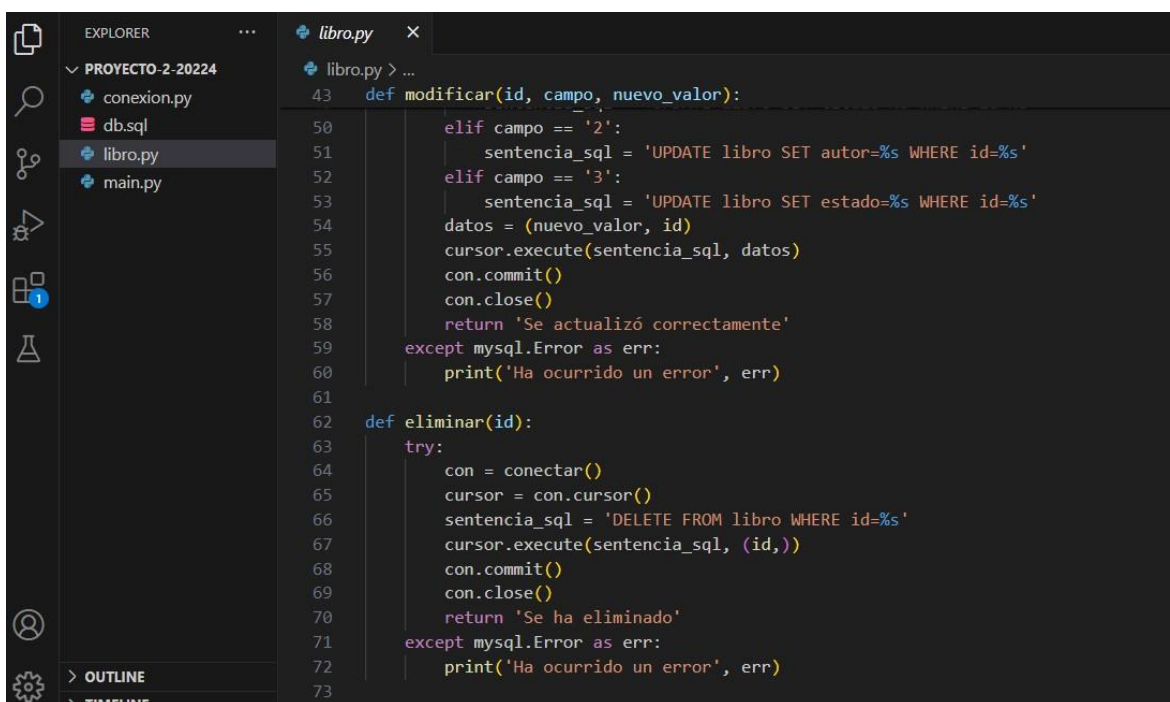


```
17 def mostrar():...
24     datos = cursor.fetchall()
25     con.close()
26 except mysql.Error as err:
27     print('Ha ocurrido un error', err)
28     return datos
29
30 def buscar(id):
31     datos = []
32     try:
33         con = conectar()
34         cursor = con.cursor()
35         sentencia_sql = "Select * from libro where id=%s"
36         cursor.execute(sentencia_sql, (id,))
37         datos = cursor.fetchall()
38         con.close()
39     except mysql.Error as err:
40         print('Ha ocurrido un error', err)
41     return datos
42
43 def modificar(id, campo, nuevo_valor):
44     try:
45         sentencia_sql = ''
46         con = conectar()
47         cursor = con.cursor()
48         if campo == '1':
```





```
43 def modificar(id, campo, nuevo_valor):
44     cursor = con.cursor()
45     if campo == '1':
46         sentencia_sql = 'UPDATE libro SET titulo=%s WHERE id=%s'
47     elif campo == '2':
48         sentencia_sql = 'UPDATE libro SET autor=%s WHERE id=%s'
49     elif campo == '3':
50         sentencia_sql = 'UPDATE libro SET estado=%s WHERE id=%s'
51     datos = (nuevo_valor, id)
52     cursor.execute(sentencia_sql, datos)
53     con.commit()
54     con.close()
55     return 'Se actualizó correctamente'
56 except mysql.Error as err:
57     print('Ha ocurrido un error', err)
58
59 def eliminar(id):
60     try:
61         con = conectar()
62         cursor = con.cursor()
63         sentencia_sql = 'DELETE FROM libro WHERE id=%s'
64         cursor.execute(sentencia_sql, (id,))
65         con.commit()
66         con.close()
67         return 'Se ha eliminado'
68     except mysql.Error as err:
69         print('Ha ocurrido un error', err)
```



```
50 elif campo == '2':
51     sentencia_sql = 'UPDATE libro SET autor=%s WHERE id=%s'
52 elif campo == '3':
53     sentencia_sql = 'UPDATE libro SET estado=%s WHERE id=%s'
54 datos = (nuevo_valor, id)
55 cursor.execute(sentencia_sql, datos)
56 con.commit()
57 con.close()
58 return 'Se actualizó correctamente'
59 except mysql.Error as err:
60     print('Ha ocurrido un error', err)
61
62 def eliminar(id):
63     try:
64         con = conectar()
65         cursor = con.cursor()
66         sentencia_sql = 'DELETE FROM libro WHERE id=%s'
67         cursor.execute(sentencia_sql, (id,))
68         con.commit()
69         con.close()
70         return 'Se ha eliminado'
71     except mysql.Error as err:
72         print('Ha ocurrido un error', err)
73
```



The image shows a code editor interface. On the left is a file explorer panel titled 'EXPLORER' with a menu icon. It shows a project named 'PROYEC...' containing three files: 'conexion.py', 'db.sql' (which is selected and highlighted in blue), 'libro.py', and 'main.py'. On the right is the editor window for 'db.sql', which contains the following SQL code:

```
1 create table libro(  
2     id integer primary key auto_increment,  
3     titulo text not null,  
4     autor text not null,  
5     estado text not null  
6 )
```