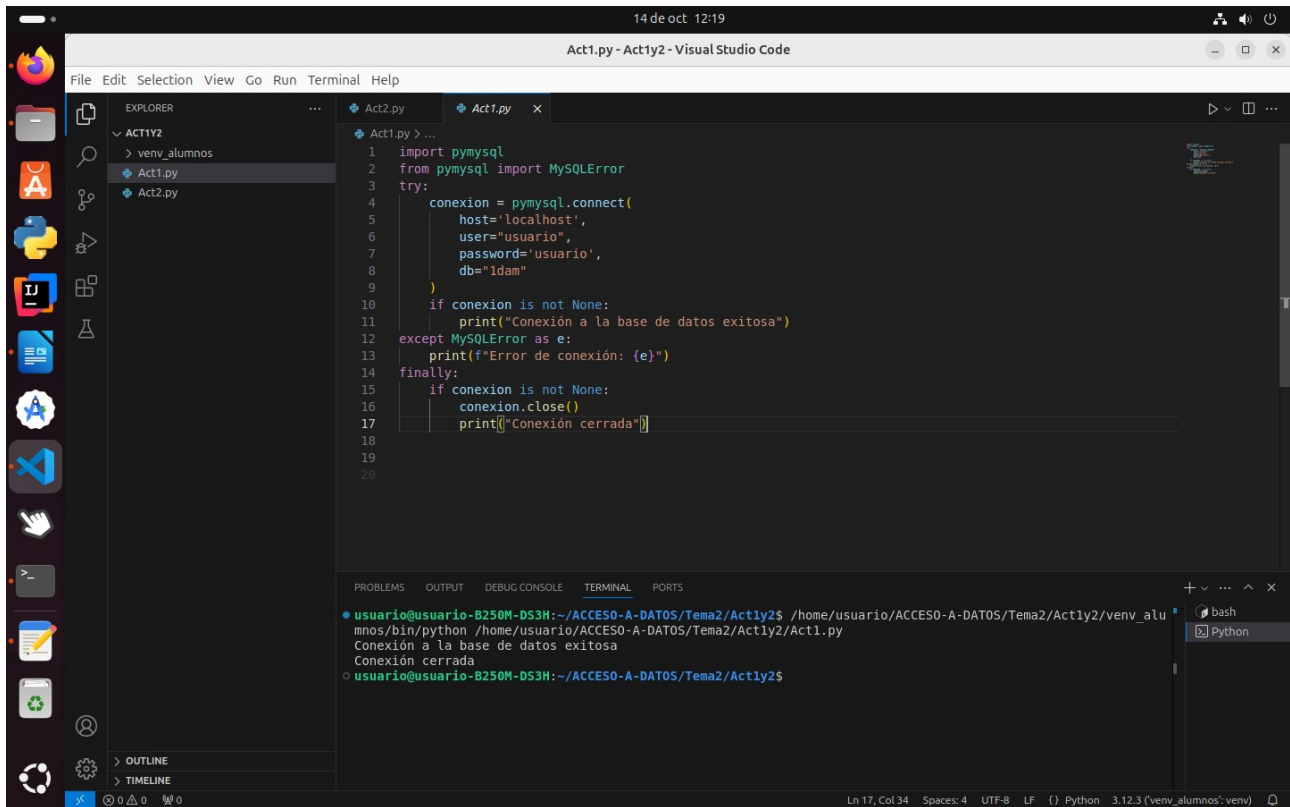


# Actividad 1



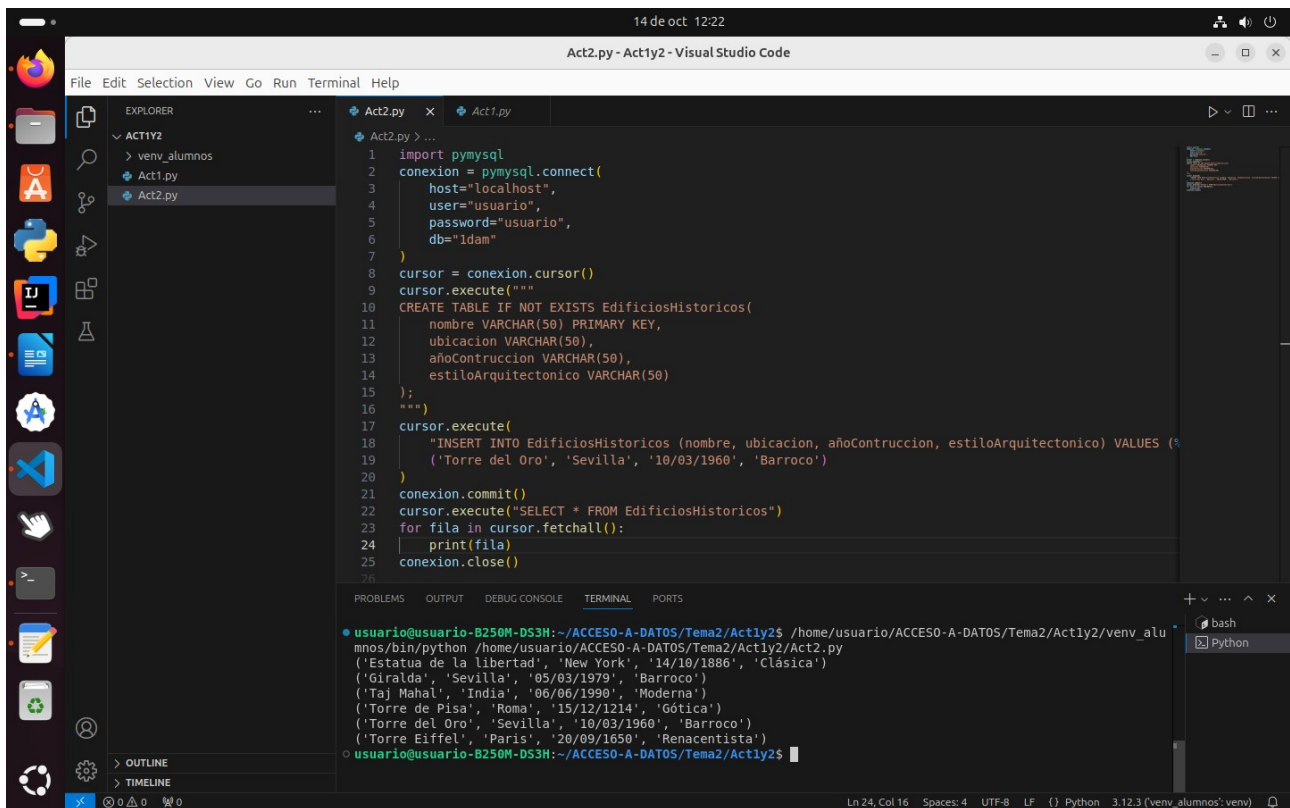
The screenshot shows the Visual Studio Code interface with the file `Act1.py` open. The code is a Python script that attempts to connect to a MySQL database. The terminal output shows the successful execution of the script, displaying the connection status.

```
1 import pymysql
2 from pymysql import MySQLError
3 try:
4     conexion = pymysql.connect(
5         host='localhost',
6         user='usuario',
7         password='usuario',
8         db='1dam'
9     )
10    if conexion is not None:
11        print("Conexión a la base de datos exitosa")
12 except MySQLError as e:
13     print(f"Error de conexión: {e}")
14 finally:
15     if conexion is not None:
16         conexion.close()
17     print("Conexión cerrada")
```

Terminal Output:

```
usuario@usuario-B250M-DS3H:~/ACCESO-A-DATOS/Tema2/Actly2$ /home/usuario/ACCESO-A-DATOS/Tema2/Actly2/venv_alu
mnos/bin/python /home/usuario/ACCESO-A-DATOS/Tema2/Actly2/Act1.py
Conexión a la base de datos exitosa
Conexión cerrada
usuario@usuario-B250M-DS3H:~/ACCESO-A-DATOS/Tema2/Actly2$
```

# Actividad 2



The screenshot shows the Visual Studio Code interface with the file `Act2.py` open. The code is a Python script that connects to a MySQL database, creates a table if it doesn't exist, inserts data, and then fetches and prints the data. The terminal output shows the successful execution of the script, displaying the connection status and the data retrieved from the database.

```
1 import pymysql
2 conexion = pymysql.connect(
3     host='localhost',
4     user='usuario',
5     password='usuario',
6     db='1dam'
7 )
8 cursor = conexion.cursor()
9 cursor.execute("""
10 CREATE TABLE IF NOT EXISTS EdificiosHistoricos(
11     nombre VARCHAR(50) PRIMARY KEY,
12     ubicacion VARCHAR(50),
13     añoConstruccion VARCHAR(50),
14     estiloArquitectonico VARCHAR(50)
15 );
16 """)
17 cursor.execute(
18     "INSERT INTO EdificiosHistoricos (nombre, ubicacion, añoConstruccion, estiloArquitectonico) VALUES (%
19     ('Torre del Oro', 'Sevilla', '18/03/1960', 'Barroco')
20 )
21 conexion.commit()
22 cursor.execute("SELECT * FROM EdificiosHistoricos")
23 for fila in cursor.fetchall():
24     print(fila)
25 conexion.close()
```

Terminal Output:

```
usuario@usuario-B250M-DS3H:~/ACCESO-A-DATOS/Tema2/Actly2$ /home/usuario/ACCESO-A-DATOS/Tema2/Actly2/venv_alu
mnos/bin/python /home/usuario/ACCESO-A-DATOS/Tema2/Actly2/Act2.py
('Estadua de la libertad', 'New York', '14/10/1886', 'Clásica')
('Giralda', 'Sevilla', '05/03/1979', 'Barroco')
('Taj Mahal', 'India', '06/06/1990', 'Moderna')
('Torre de Pisa', 'Roma', '15/12/1214', 'Gótica')
('Torre del Oro', 'Sevilla', '18/03/1960', 'Barroco')
('Torre Eiffel', 'Paris', '28/09/1650', 'Renacentista')
usuario@usuario-B250M-DS3H:~/ACCESO-A-DATOS/Tema2/Actly2$
```

## Actividad 3

Tiempo de inserción con mysql-connector: 1.4116413593292236 segundos

Tiempo de inserción con mysql-connector: 1.3135454654693604 segundos

Tiempo de lectura con mysql-connector: 1.3494782447814941 segundos

Tiempo de lectura con mysql-connector: 1.3893604278564453 segundos

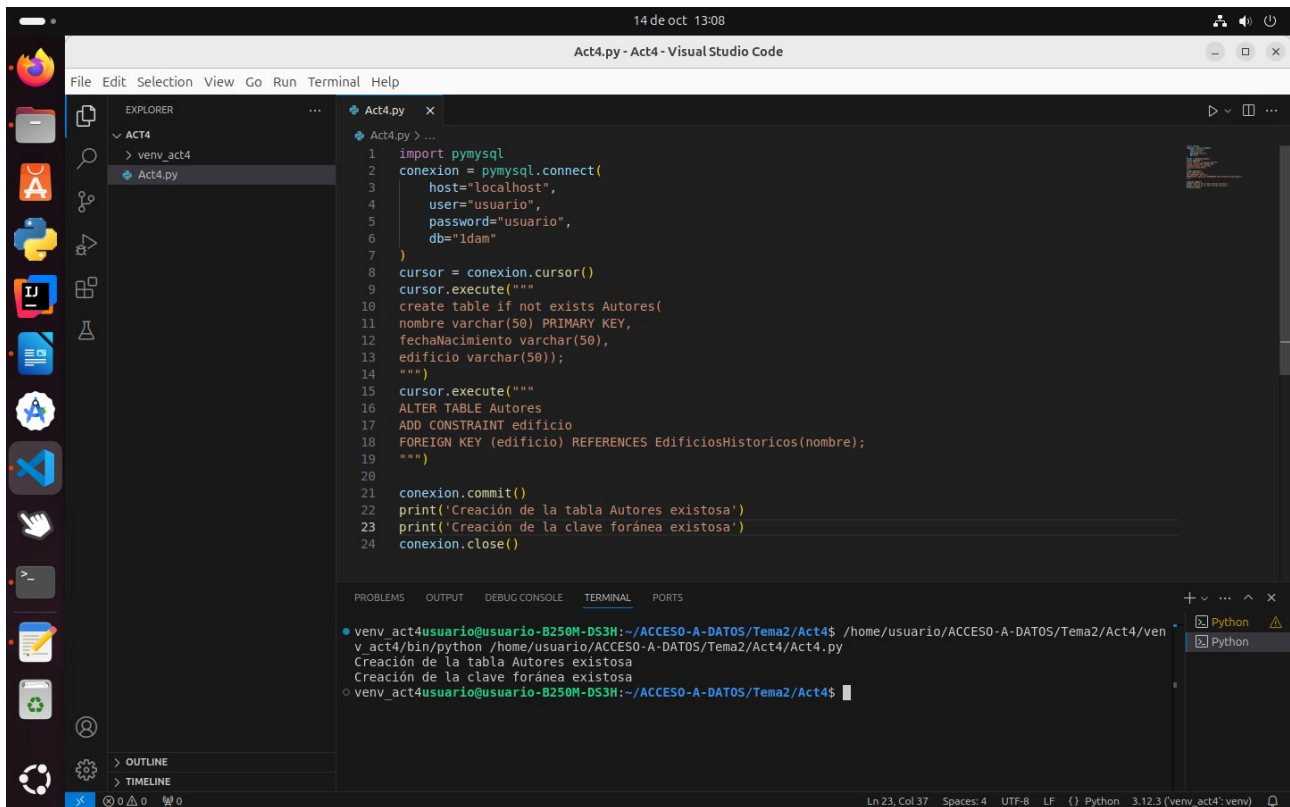
Tiempo de inserción con PyMySQL: 1.1587028503417969 segundos

Tiempo de inserción con PyMySQL: 1.1890358924865723 segundos

Tiempo de lectura con PyMySQL: 0.004366397857666016 segundos

Tiempo de lectura con PyMySQL: 0.004811286926269531 segundos

## Actividad 4



The screenshot shows the Visual Studio Code interface with a file named `Act4.py` open. The code in the editor is as follows:

```
1 import pymysql
2 conexion = pymysql.connect(
3     host="localhost",
4     user="usuario",
5     password="usuario",
6     db="ldam"
7 )
8 cursor = conexion.cursor()
9 cursor.execute("""
10 create table if not exists Autores(
11     nombre varchar(50) PRIMARY KEY,
12     fechaNacimiento varchar(50),
13     edificio varchar(50));
14 """)
15 cursor.execute("""
16 ALTER TABLE Autores
17 ADD CONSTRAINT edificio
18 FOREIGN KEY (edificio) REFERENCES EdificiosHistoricos(nombre);
19 """)
20
21 conexion.commit()
22 print('Creación de la tabla Autores existosa')
23 print('Creación de la clave foránea existosa')
24 conexion.close()
```

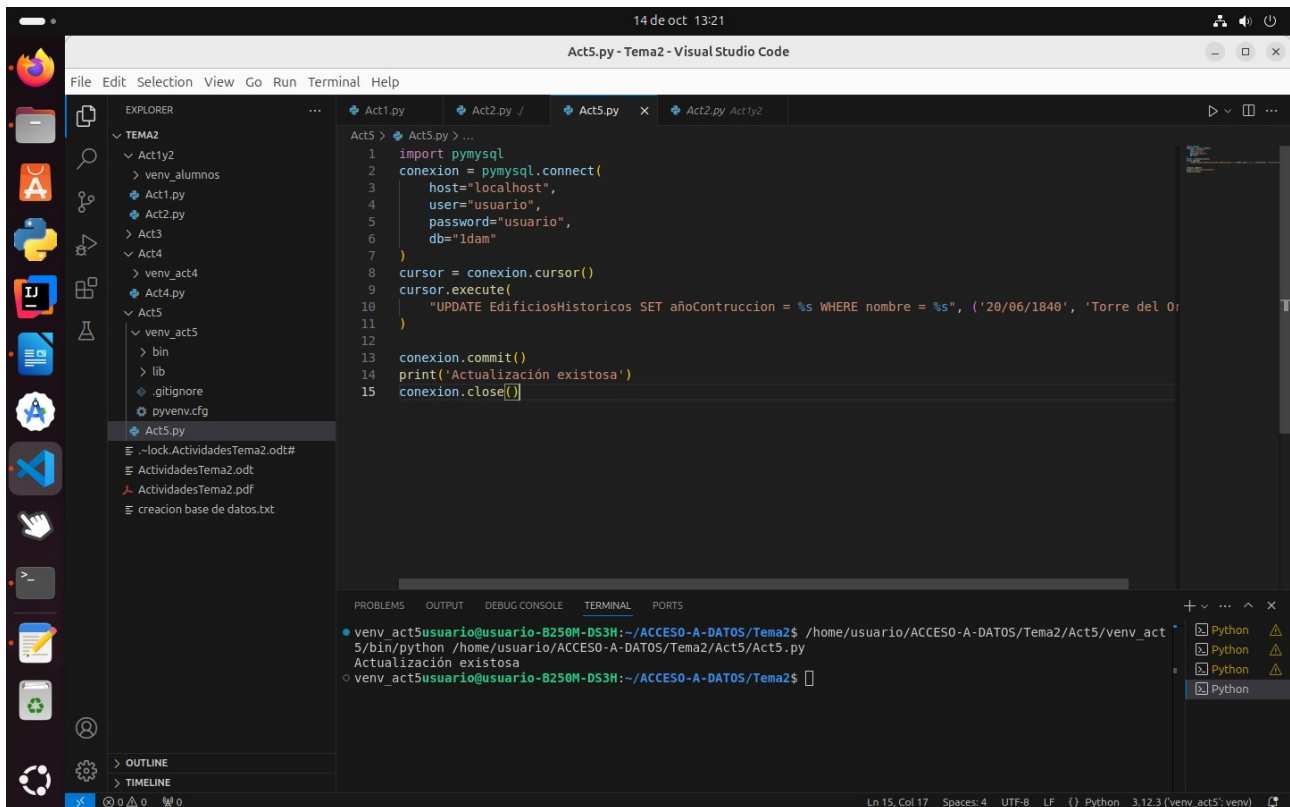
Below the code editor, the terminal window shows the command to run the script and its output:

```
● venv act4usuario@usuario-B250M-D53H:~/ACCESO-A-DATOS/Tema2/Act4$ /home/usuario/ACCESO-A-DATOS/Tema2/Act4/ven
v_act4/bin/python /home/usuario/ACCESO-A-DATOS/Tema2/Act4/Act4.py
Creación de la tabla Autores existosa
Creación de la clave foránea existosa
○ venv_act4usuario@usuario-B250M-D53H:~/ACCESO-A-DATOS/Tema2/Act4$
```

The status bar at the bottom indicates the current line and column: `Ln 23, Col 37`, along with other settings like `Spaces: 4`, `UTF-8`, `LF`, and the Python version `3.12.3 (venv_act4: venv)`.

## Actividad 5

- Actuaización:



```
Act5.py - Tema2 - Visual Studio Code

1 import pymysql
2 conexion = pymysql.connect(
3     host="localhost",
4     user="usuario",
5     password="usuario",
6     db="ldam"
7 )
8 cursor = conexion.cursor()
9 cursor.execute(
10     "UPDATE EdificiosHistoricos SET añoContruccion = %s WHERE nombre = %s", ('20/06/1840', 'Torre del Oro')
11 )
12
13 conexion.commit()
14 print('Actualización exitosa')
15 conexion.close()
```

```
venv_act5usuario@usuario-B250M-D53H:~/ACCESO-A-DATOS/Tema2$ /home/usuario/ACCESO-A-DATOS/Tema2/Act5/venv_act5/bin/python /home/usuario/ACCESO-A-DATOS/Tema2/Act5/Act5.py
Actualización exitosa
venv_act5usuario@usuario-B250M-D53H:~/ACCESO-A-DATOS/Tema2$
```

```
+-----+-----+-----+-----+
| Estatua de la libertad | New York | 14/10/1886 | Clásica |
| Giralda                | Sevilla | 05/03/1979 | Barroco |
| Taj Mahal              | India   | 06/06/1990 | Moderna |
| Torre de Pisa          | Roma    | 15/12/1214 | Gótica  |
| Torre del Oro          | Sevilla | 10/03/1960 | Barroco |
| Torre Eiffel           | Paris   | 20/09/1650 | Renacentista |
+-----+-----+-----+-----+
6 rows in set (0,00 sec)

mysql> select * from EdificiosHistoricos;
+-----+-----+-----+-----+
| nombre                | ubicacion | añoContruccion | estiloArquitectonico |
+-----+-----+-----+-----+
| Estatua de la libertad | New York  | 14/10/1886     | Clásica               |
| Giralda                | Sevilla  | 05/03/1979     | Barroco               |
| Taj Mahal              | India    | 06/06/1990     | Moderna               |
| Torre de Pisa          | Roma     | 15/12/1214     | Gótica                |
| Torre del Oro          | Sevilla  | 20/06/1840     | Barroco               |
| Torre Eiffel           | Paris    | 20/09/1650     | Renacentista          |
+-----+-----+-----+-----+
```

- Eliminación:

Visual Studio Code interface showing the file explorer on the left with the project structure. The main editor displays the 'Delete.py' file. The terminal at the bottom shows the execution of the script, which successfully deletes a record from the 'EdificiosHistoricos' table.

```

1 import pymysql
2 conexion = pymysql.connect(
3     host="localhost",
4     user="usuario",
5     password="usuario",
6     db="ldam"
7 )
8 cursor = conexion.cursor()
9
10 cursor.execute(
11     "DELETE FROM EdificiosHistoricos WHERE nombre = %s", ('Torre del Oro',)
12 )
13
14 conexion.commit()
15 print('Eliminación exitosa')
16 conexion.close()

```

Terminal output:

```

venv_act5usuario@usuario-B250M-D53H:~/ACCESO-A-DATOS/Tema2$ /home/usuario/ACCESO-A-DATOS/Tema2/Act5/venv_act5/bin/python /home/usuario/ACCESO-A-DATOS/Tema2/Act5/Delete.py
Eliminación exitosa
venv_act5usuario@usuario-B250M-D53H:~/ACCESO-A-DATOS/Tema2$

```

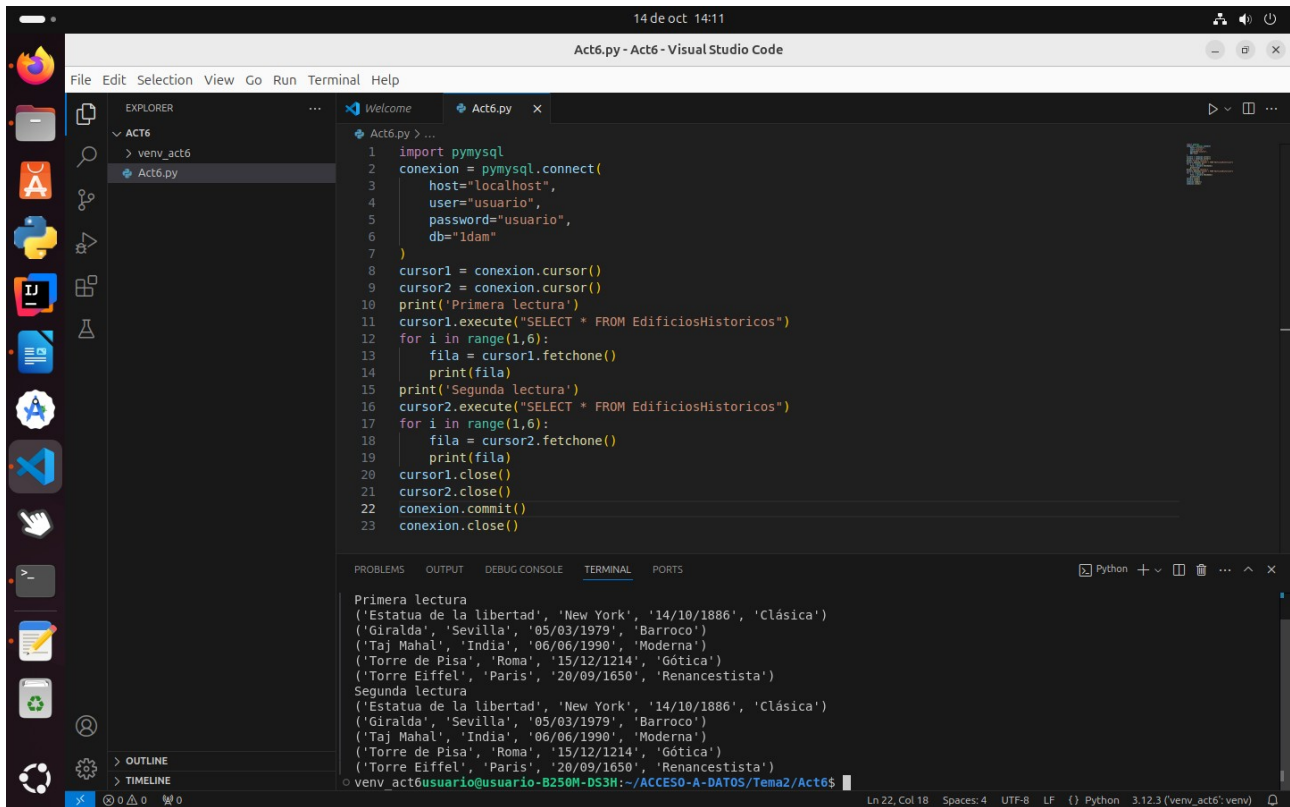
nombre	ubicacion	añoContruccion	estiloArquitectonico
Estatua de la libertad	New York	14/10/1886	Clásica
Giralda	Sevilla	05/03/1979	Barroco
Taj Mahal	India	06/06/1990	Moderna
Torre de Pisa	Roma	15/12/1214	Gótica
Torre del Oro	Sevilla	20/06/1840	Barroco
Torre Eiffel	Paris	20/09/1650	Renancestista

6 rows in set (0,00 sec)

```
mysql> select * from EdificiosHistoricos;
```

nombre	ubicacion	añoContruccion	estiloArquitectonico
Estatua de la libertad	New York	14/10/1886	Clásica
Giralda	Sevilla	05/03/1979	Barroco
Taj Mahal	India	06/06/1990	Moderna
Torre de Pisa	Roma	15/12/1214	Gótica
Torre Eiffel	Paris	20/09/1650	Renancestista

## Actividad 6



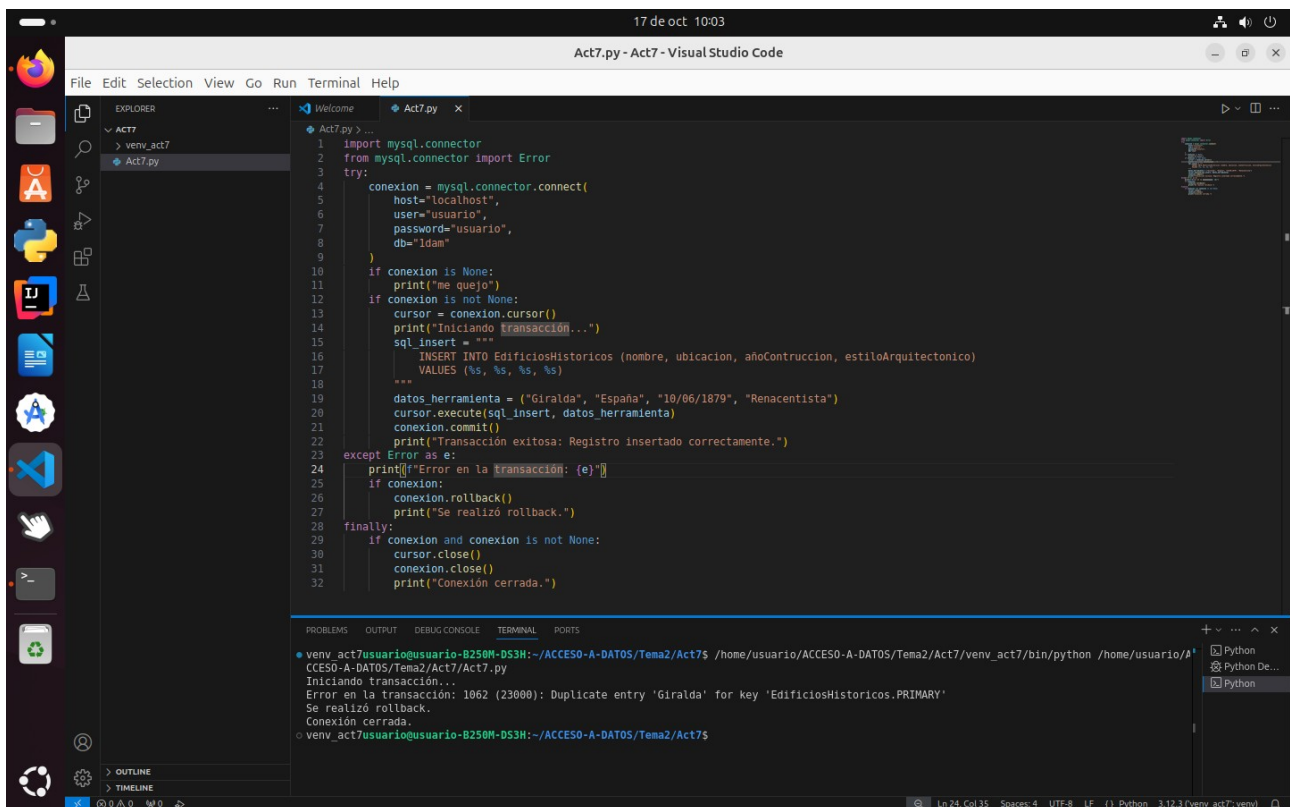
The screenshot shows the Visual Studio Code editor with a file named `Act6.py` open. The code is a Python script that connects to a MySQL database and prints two sets of data from a table named `EdificiosHistoricos`. The terminal output shows the results of the queries.

```
1 import pymysql
2 conexion = pymysql.connect(
3     host="localhost",
4     user="usuario",
5     password="usuario",
6     db="ldam"
7 )
8 cursor1 = conexion.cursor()
9 cursor2 = conexion.cursor()
10 print('Primera lectura')
11 cursor1.execute("SELECT * FROM EdificiosHistoricos")
12 for i in range(1,6):
13     fila = cursor1.fetchone()
14     print(fila)
15 print('Segunda lectura')
16 cursor2.execute("SELECT * FROM EdificiosHistoricos")
17 for i in range(1,6):
18     fila = cursor2.fetchone()
19     print(fila)
20 cursor1.close()
21 cursor2.close()
22 conexion.commit()
23 conexion.close()
```

Terminal Output:

```
Primera lectura
('Estatua de la libertad', 'New York', '14/10/1886', 'Clásica')
('Giralda', 'Sevilla', '05/03/1979', 'Barroco')
('Taj Mahal', 'India', '06/06/1990', 'Moderna')
('Torre de Pisa', 'Roma', '15/12/1214', 'Gótica')
('Torre Eiffel', 'Paris', '20/09/1650', 'Renacentista')
Segunda lectura
('Estatua de la libertad', 'New York', '14/10/1886', 'Clásica')
('Giralda', 'Sevilla', '05/03/1979', 'Barroco')
('Taj Mahal', 'India', '06/06/1990', 'Moderna')
('Torre de Pisa', 'Roma', '15/12/1214', 'Gótica')
('Torre Eiffel', 'Paris', '20/09/1650', 'Renacentista')
```

## Actividad 7



The screenshot shows the Visual Studio Code editor with a file named `Act7.py` open. The code is a Python script that connects to a MySQL database and inserts a new record into the `EdificiosHistoricos` table. The terminal output shows the results of the execution, including an error message.

```
1 import mysql.connector
2 from mysql.connector import Error
3 try:
4     conexion = mysql.connector.connect(
5         host="localhost",
6         user="usuario",
7         password="usuario",
8         db="ldam"
9     )
10 if conexion is None:
11     print("me quejo")
12 if conexion is not None:
13     cursor = conexion.cursor()
14     print("Iniciando transacción...")
15     sql_insert = """
16         INSERT INTO EdificiosHistoricos (nombre, ubicacion, añoConstruccion, estiloArquitectonico)
17         VALUES (%s, %s, %s, %s)
18     """
19     datos_herramienta = ("Giralda", "España", "10/06/1879", "Renacentista")
20     cursor.execute(sql_insert, datos_herramienta)
21     conexion.commit()
22     print("Transacción exitosa: Registro insertado correctamente.")
23 except Error as e:
24     print(f"Error en la transacción: {e}")
25     if conexion:
26         conexion.rollback()
27     print("Se realizó rollback.")
28 finally:
29     if conexion and conexion is not None:
30         cursor.close()
31         conexion.close()
32     print("Conexión cerrada.")
```

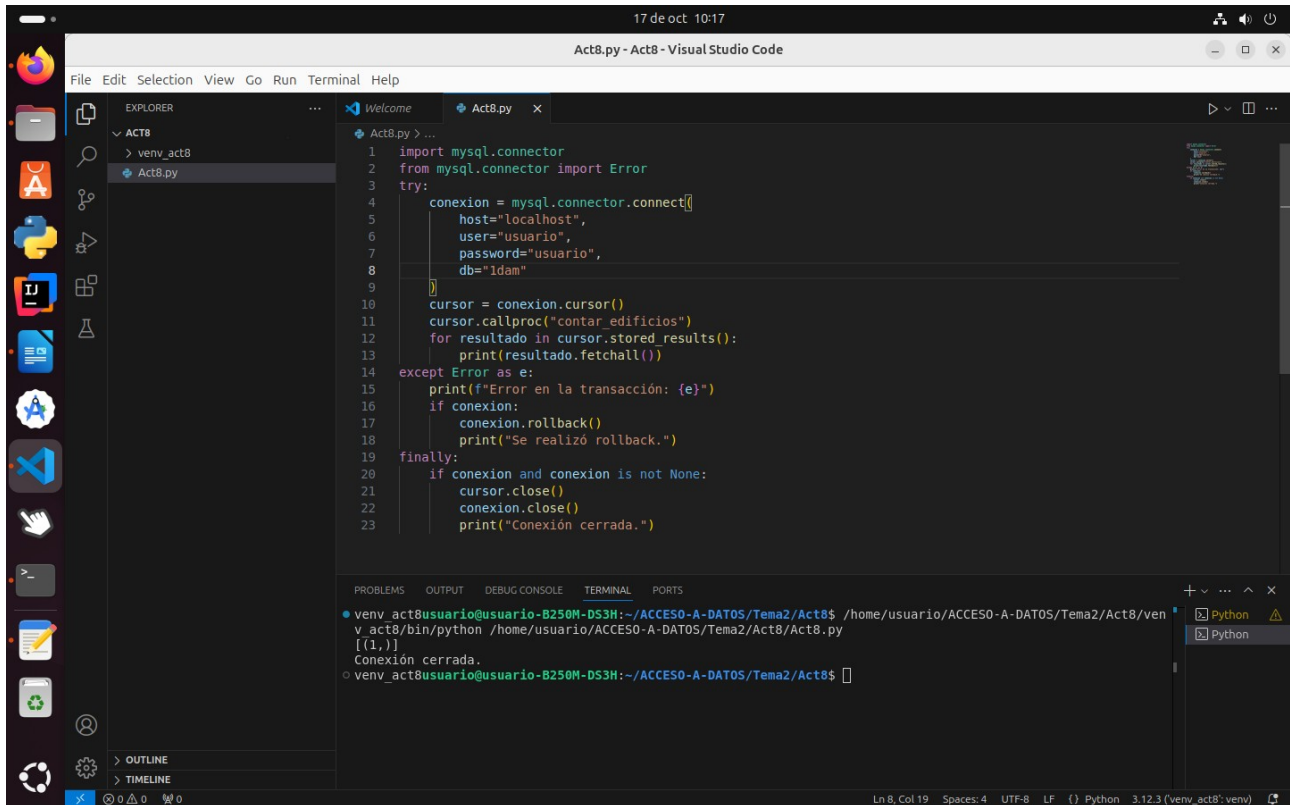
Terminal Output:

```
venv_act7usuario@usuario-B250M-D53H:~/ACCESO-A-DATOS/Tema2/Act7$ python /home/usuario/ACCESO-A-DATOS/Tema2/Act7/venv_act7/bin/python /home/usuario/ACCESO-A-DATOS/Tema2/Act7/Act7.py
Iniciando transacción...
Error en la transacción: 1062 (23000): Duplicate entry 'Giralda' for key 'EdificiosHistoricos.PRIMARY'
Se realizó rollback.
Conexión cerrada.
```



## Actividad 8

```
mysql> CREATE PROCEDURE contar_edificios() BEGIN select count(*) from EdificiosH  
istoricos where estiloArquitectonico="Barroco"; END//  
Query OK, 0 rows affected (0,01 sec)  
  
mysql> delimiter ;  
mysql> 
```



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows a project named 'ACT8' with a subdirectory 'venv\_act8' containing a file 'Act8.py'. The main editor displays the code for 'Act8.py', which is a Python script that connects to a MySQL database, calls a stored procedure, and prints the results. The script includes error handling and a rollback mechanism. The bottom panel shows the TERMINAL output, which displays the command to run the script and the resulting output: a list containing the number 1, followed by a message indicating the connection is closed.

```
17 de oct 10:17  
Act8.py - Act8 - Visual Studio Code  
File Edit Selection View Go Run Terminal Help  
EXPLORER  
  ACT8  
  > venv_act8  
    Act8.py  
  Act8.py > ...  
1  import mysql.connector  
2  from mysql.connector import Error  
3  try:  
4      conexion = mysql.connector.connect(  
5          host="localhost",  
6          user="usuario",  
7          password="usuario",  
8          db="1dam"  
9      )  
10     cursor = conexion.cursor()  
11     cursor.callproc("contar_edificios")  
12     for resultado in cursor.stored_results():  
13         print(resultado.fetchall())  
14 except Error as e:  
15     print(f"Error en la transacción: {e}")  
16     if conexion:  
17         conexion.rollback()  
18         print("Se realizó rollback.")  
19 finally:  
20     if conexion and conexion is not None:  
21         cursor.close()  
22         conexion.close()  
23     print("Conexión cerrada.")  
  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
• venv_act8usuario@usuario-B250M-D53H:~/ACCESO-A-DATOS/Tema2/Act8$ /home/usuario/ACCESO-A-DATOS/Tema2/Act8/ven  
v_act8/bin/python /home/usuario/ACCESO-A-DATOS/Tema2/Act8/Act8.py  
[[1,]]  
Conexión cerrada.  
o venv_act8usuario@usuario-B250M-D53H:~/ACCESO-A-DATOS/Tema2/Act8$  
Ln 8, Col 19 Spaces: 4 UTF-8 LF Python 3.12.3 (venv_act8: venv)
```