



📌 Cómo eliminar un registro en PostgreSQL utilizando Python

🌟 Introducción

En esta guía aprenderás cómo **eliminar un registro específico** de una base de datos PostgreSQL utilizando Python. Para ello, utilizaremos la sentencia **DELETE**, acompañada de un filtro para indicar qué registro eliminar. Esta operación es fundamental para mantener actualizada y limpia la base de datos.

🔥 Paso 1: Conexión y apertura de cursor

📁 **Archivo:** prueba_bd.py

Descripción:

Primero establecemos la conexión con la base de datos y abrimos un cursor para ejecutar las instrucciones SQL.

```
import psycopg
```

```
with psycopg.connect(
```

```
user="postgres", password="admin",  
host="localhost", port="5432", dbname="test_db"  
) as conexion:  
    with conexion.cursor() as cursor:
```

👉 Explicación:

El uso de `with` asegura que la conexión y el cursor se cierren automáticamente al finalizar su uso.

🔥 Paso 2: Definir la sentencia SQL de eliminación

Descripción:

La sentencia **DELETE** indica que se eliminará un registro de la tabla **persona**, especificando un filtro por **id_persona**.

```
sentencia = "DELETE FROM persona WHERE id_persona=%s"
```

👉 Explicación:

El parámetro `%s` es un marcador de posición (placeholder) que será sustituido con el valor proporcionado.

🔥 Paso 3: Obtener el id de persona a eliminar

Descripción:

Solicitamos al usuario que ingrese el **id_persona** que desea eliminar y lo almacenamos en una tupla.

```
entrada = input("Proporciona el id_persona a eliminar: ")  
valores = (entrada,)
```

👉 Explicación:

El valor ingresado por el usuario se convierte en una tupla con una sola posición, necesario para que sea compatible con la ejecución parametrizada.

🔥 Paso 4: Ejecutar la sentencia SQL

Descripción:

Usamos el método **execute** para ejecutar la instrucción SQL, pasando la sentencia y la tupla de valores.

```
cursor.execute(sentencia, valores)
```

👉 Explicación:

La ejecución reemplaza de forma segura el marcador `%s` con el valor de `id_persona` proporcionado.

🔥 Paso 5: Mostrar cantidad de registros eliminados

Descripción:

Consultamos cuántos registros fueron eliminados usando `rowcount`.

```
registros_eliminados = cursor.rowcount
print(f"Registros eliminados: {registros_eliminados}")
```

👉 Explicación:

`cursor.rowcount` devuelve cuántos registros fueron afectados por la operación. Si el valor es **0**, significa que no se eliminó ningún registro porque no existía ese `id_persona`.

✅ Código completo por archivo

📁 Archivo completo: `prueba_bd.py`

➡ Código final completo de este archivo.

```
import psycopg

with psycopg.connect(
    user="postgres", password="admin",
    host="localhost", port="5432", dbname="test_db"
) as conexion:
    with conexion.cursor() as cursor:
        sentencia = "DELETE FROM persona WHERE id_persona=%s"
        entrada = input("Proporciona el id_persona a eliminar: ")
        valores = (entrada,)
        cursor.execute(sentencia, valores)
        registros_eliminados = cursor.rowcount
        print(f"Registros eliminados: {registros_eliminados}")
```

🎉 Conclusión

En esta guía aprendiste a **eliminar un registro específico** de una base de datos PostgreSQL mediante Python. Usamos una sentencia **DELETE** con un parámetro seguro y solicitamos al usuario el **id_persona** que desea eliminar. Este enfoque es seguro y controlado, evitando eliminar registros por accidente.

Con este conocimiento podrás gestionar la eliminación de registros de manera interactiva y eficiente. 🚀

✨ Sigue adelante con tu aprendizaje 🚀 , ¡el esfuerzo vale la pena!

¡Saludos! 🙌

Ing. Marcela Gamiño e Ing. Ubaldo Acosta

Fundadores de [GlobalMentoring.com.mx](https://www.globalmentoring.com.mx)