



## 📌 Cómo eliminar varios registros en PostgreSQL utilizando Python

---

### ✨ Introducción

En esta guía aprenderás cómo **eliminar varios registros a la vez** en una base de datos PostgreSQL utilizando Python. Usaremos la sentencia **DELETE** con la cláusula **IN**, lo cual nos permitirá eliminar múltiples registros especificando sus claves primarias.

---

### 🔥 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` garantiza que la conexión y el cursor se cierren automáticamente al finalizar.

---

## 🔥 Paso 2: Obtener los `id_persona` a eliminar

#### Descripción:

Solicitamos al usuario los `id_persona` a eliminar, separados por comas, y los convertimos en una tupla.

```
entrada = input("Proporciona los id_persona a eliminar separados por coma: ")  
valores = tuple(entrada.split(','))
```

#### 👉 Explicación:

Usamos `split(',')` para separar los valores y luego los convertimos en una tupla que será pasada a la consulta SQL.

---

## 🔥 Paso 3: Definir la sentencia SQL de eliminación múltiple

#### Descripción:

La sentencia `DELETE` usará la cláusula `IN` para eliminar múltiples registros según sus `id_persona`.

```
placeholders = ','.join(['%s'] * len(valores))  
sentencia = f"DELETE FROM persona WHERE id_persona IN ({placeholders})"
```

#### 👉 Explicación:

Generamos los `placeholders` **dinámicamente** según la cantidad de valores que se ingresen. Esto permite construir la consulta correctamente para varios `id_persona`.

---

## 🔥 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 los **placeholders** de manera segura con los valores proporcionados.

---

## 🔥 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 eliminados efectivamente. Si no se encuentra algún `id_persona`, simplemente no lo elimina.

---

## ✅ Código completo por archivo

📁 Archivo completo: `prueba_bd.py`

➡ Código final completo de este archivo.

```
import psycopgp

with psycopgp.connect(
    user="postgres", password="admin",
    host="localhost", port="5432", dbname="test_db"
) as conexion:
    with conexion.cursor() as cursor:
        entrada = input("Proporciona los id_persona a eliminar separados por coma: ")
        valores = tuple(entrada.split(','))
        placeholders = ','.join(['%s'] * len(valores))
        sentencia = f"DELETE FROM persona WHERE id_persona IN ({placeholders})"
        cursor.execute(sentencia, valores)
        registros_eliminados = cursor.rowcount
        print(f"Registros eliminados: {registros_eliminados}")
```

---

## Conclusión

En esta guía aprendiste a **eliminar múltiples registros a la vez** en una base de datos PostgreSQL utilizando Python. Aplicamos una sentencia **DELETE** con la cláusula **IN** y parametrización segura, permitiendo eliminar varios registros de manera sencilla y controlada.

Este enfoque facilita la gestión masiva de datos y evita errores al eliminar registros uno por uno. 🚀

---

💎 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)