




## Guía: Implementación del método eliminar en PersonaDAO (DAO) con Python y PostgreSQL

### Introducción

En esta lección implementaremos el método `eliminar` dentro de la clase `PersonaDAO`. Este método permite borrar registros de la base de datos utilizando el patrón DAO. Además, realizaremos una prueba para confirmar que el objeto `Persona` se elimina correctamente con base en su ID.

---

### Paso 1: Ubicación del archivo trabajado

 **Ruta del archivo:** `capa_datos_persona/persona_dao.py`

Este archivo contiene la definición de la clase `PersonaDAO` y sus métodos CRUD.

---

### Paso 2: Agregar el método `eliminar`

#### Descripción

Se implementa el método `eliminar`, que recibe un objeto `Persona` y elimina el registro correspondiente de la base de datos con base en su `id_persona`.

### Código trabajado (`capa_datos_persona/persona_dao.py`):

```
@classmethod
def eliminar(cls, persona):
    with Conexion.obtener_conexion() as conexion:
        with conexion.cursor() as cursor:
            valores = (persona.id_persona,)
            cursor.execute(cls._ELIMINAR, valores)
            log.debug(f'Objeto eliminado: {persona}')
            return cursor.rowcount
```

### Explicación:

- Se usa el atributo `_ELIMINAR` con la sentencia SQL `DELETE`.
- Se abre una conexión y cursor dentro de un bloque `with`.
- Se construye una tupla con el `id_persona`, **agregando una coma al final** para que se interprete correctamente como tupla.
- Se ejecuta el método `execute()` con la sentencia SQL y los valores.
- Se registra la eliminación en el log.
- Se retorna el número de registros eliminados con `cursor.rowcount`.

## Paso 3: Probar el método `eliminar`

### Descripción

Agregamos una prueba en el bloque `main` del mismo archivo para eliminar un registro de la tabla `persona`.

### Código trabajado (`capa_datos_persona/persona_dao.py`):


```
if __name__ == '__main__':
    # Eliminar un registro
    persona1 = Persona(id_persona=7)
    personas_eliminadas = PersonaDAO.eliminar(persona1)
    log.debug(f'Personas eliminadas: {personas_eliminadas}')


    # Seleccionar objetos
    personas = PersonaDAO.seleccionar()
    for persona in personas:
        log.debug(persona)
```

**✓ Explicación:**

- Creamos un objeto `Persona` con solo el `id_persona` especificado.
- Llamamos a `PersonaDAO.eliminar()` pasando el objeto a eliminar.
- Registramos cuántas personas se eliminaron.
- Usamos `seleccionar()` para mostrar los registros restantes y confirmar la eliminación.

**Sección final: Código completo de los archivos trabajados**

 Aquí puedes agregar el código completo para el alumno:

 **Ruta y nombre del archivo:** `capa_datos_persona/persona_dao.py`

```
from conexion import Conexion

from persona import Persona
from logger_base import log

class PersonaDAO:
    """
    DAO (Data Access Object)
    CRUD (Create-Read-Update-Delete)
    """
    _SELECCIONAR = 'SELECT * FROM persona ORDER BY id_persona'
    _INSERTAR = 'INSERT INTO persona(nombre, apellido, email) VALUES(%s, %s, %s)'
    _ACTUALIZAR = 'UPDATE persona SET nombre=%s, apellido=%s, email=%s WHERE id_persona=%s'
    _ELIMINAR = 'DELETE FROM persona WHERE id_persona=%s'

    @classmethod
    def seleccionar(cls):
        with Conexion.obtener_conexion() as conexion:
            with conexion.cursor() as cursor:
                cursor.execute(cls._SELECCIONAR)
                registros = cursor.fetchall()
                personas = []
                for registro in registros:
                    persona = Persona(registro[0], registro[1], registro[2], registro[3])
                    personas.append(persona)
                return personas

    @classmethod
    def insertar(cls, persona):
        with Conexion.obtener_conexion() as conexion:
            with conexion.cursor() as cursor:
                valores = (persona.nombre, persona.apellido, persona.email)
```

```
        cursor.execute(cls._INSERTAR, valores)
        log.debug(f'Persona insertada: {persona}')
        return cursor.rowcount

    @classmethod
    def actualizar(cls, persona):
        with Conexion.obtener_conexion() as conexion:
            with conexion.cursor() as cursor:
                valores = (persona.nombre, persona.apellido, persona.email, persona.id_persona)
                cursor.execute(cls._ACTUALIZAR, valores)
                log.debug(f'Persona actualizada: {persona}')
                return cursor.rowcount

    @classmethod
    def eliminar(cls, persona):
        with Conexion.obtener_conexion() as conexion:
            with conexion.cursor() as cursor:
                valores = (persona.id_persona,)
                cursor.execute(cls._ELIMINAR, valores)
                log.debug(f'Objeto eliminado: {persona}')
                return cursor.rowcount

if __name__ == '__main__':
    # Insertar un registro
    # persona1 = Persona(nombre='Pedro', apellido='Najera', email='pnajera@mail.com')
    # personas_insertadas = PersonaDAO.insertar(persona1)
    # log.debug(f'Personas insertadas: {personas_insertadas}')

    # Actualizar un registro
    # persona1 = Persona(1, 'Juan Carlos', 'Juarez', 'cjjuarez@mail.com')
    # personas_actualizadas = PersonaDAO.actualizar(persona1)
    # log.debug(f'Personas actualizadas: {personas_actualizadas}')

    # Eliminar un registro
    persona1 = Persona(id_persona=7)
    personas_eliminadas = PersonaDAO.eliminar(persona1)
    log.debug(f'Personas eliminadas: {personas_eliminadas}')

    # Seleccionar objetos
    personas = PersonaDAO.seleccionar()
    for persona in personas:
        log.debug(persona)
```

---



## Conclusión



¡Has completado la implementación del método `eliminar` usando el patrón DAO en Python! Este método permite eliminar registros de manera segura y reutilizable, y completa el conjunto de operaciones CRUD (Crear, Leer, Actualizar y Eliminar).



Más adelante se mejorará aún más esta estructura utilizando un **pool de conexiones**, lo que hará que la gestión de recursos sea más 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)