

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
FACULTAD DE CIENCIAS, 2025-II  
FUNDAMENTOS DE BASES DE DATOS



---

## PRACTICA06

### *Mantenimiento de Llaves Foráneas.*

---

NOMBRE DEL EQUIPO:  
NAMEisNULL

INTEGRANTES:

Flores Mata Ricardo - 422127808

Matute Cantón Sara Lorena - 319331622

Martinez Leal Jose Maria - 317243970

Sánchez Cruz Norma Selene - 320198508

Suárez Ortiz Joshua Daniel - 320151260

Villegas Martínez Vania Victoria - 418003114

## 1. ¿Qué es una política de mantenimiento de llaves foráneas?

Una política de mantenimiento de llaves foráneas define el comportamiento de la base de datos cuando se elimina o actualiza un registro de una tabla que está siendo referenciado por otra. Estas políticas permiten mantener la integridad referencial entre tablas, evitando inconsistencias y errores en los datos [1].

## 2. ¿Cómo se indica en SQL cada política?

En PostgreSQL las políticas de mantenimiento se especifican en la definición de las llaves foráneas usando las sentencias:

```
FOREIGN KEY (columna_hija)
REFERENCES tabla_padre (columna_padre)
ON DELETE acción
ON UPDATE acción;
```

Las acciones posibles son:

- CASCADE
- SET NULL
- SET DEFAULT
- RESTRICT
- NO ACTION

## 3. ¿Cuál es el objeto y funcionamiento de cada política?

- **CASCADE**: Cuando se elimina o actualiza un registro padre, automáticamente se eliminan o actualizan los registros hijos [3].
- **SET NULL**: Al eliminar o actualizar un registro padre, los campos de la llave foránea en los hijos se establecen en NULL.
- **SET DEFAULT**: Similar a SET NULL, pero asigna el valor por defecto de la columna.
- **RESTRICT**: Impide la eliminación o actualización del registro padre si existen registros hijos relacionados.
- **NO ACTION**: Igual que RESTRICT, pero la comprobación ocurre al final de la transacción.

## 4. Ventajas y desventajas de cada política

- **CASCADE**
  - Ventaja: Automatiza mantenimiento y evita registros huérfanos.
  - Desventaja: Riesgo de eliminar muchos registros sin darse cuenta.
- **SET NULL**
  - Ventaja: Permite conservar registros hijos.
  - Desventaja: Puede dejar registros incompletos.
- **SET DEFAULT**
  - Ventaja: Útil si existe un valor por defecto representativo.
  - Desventaja: Requiere que la columna tenga un valor default definido.
- **RESTRICT**
  - Ventaja: Protege los datos de ser eliminados por accidente.
  - Desventaja: Requiere eliminar o actualizar primero las filas hijas.
- **NO ACTION**
  - Ventaja: Permite trabajar con transacciones complejas.

- Desventaja: Es más complicado de entender o predecir su efecto.

## 5. Política seleccionada y justificación

Para el esquema que estamos trabajando, se utilizarán dos políticas de mantenimiento de llaves foráneas: **ON DELETE CASCADE** y **ON UPDATE CASCADE**.

La política **ON DELETE CASCADE** se utilizará porque resulta conveniente que, al eliminar un registro de la tabla padre (por ejemplo, un **Bazar**), también se eliminen automáticamente todas las filas relacionadas en las tablas hijas (como **AmenidadBazar**). Esto evita que existan registros huérfanos y simplifica el mantenimiento de la base de datos, asegurando que los datos mantengan coherencia e integridad.

Por otro lado, también se empleará la política **ON UPDATE CASCADE**, ya que de esta manera, si por alguna razón se modifica el valor del identificador principal (**IdBazar**) en la tabla padre, este cambio se propagará automáticamente a las tablas hijas que dependan de dicho identificador. Aunque no es común actualizar los valores de las llaves primarias, esta política permite mantener la integridad referencial de forma automática en caso de que esta situación ocurra.

En resumen, ambas políticas en conjunto permiten:

- Facilitar el mantenimiento de los datos.
- Evitar registros huérfanos.
- Asegurar que los cambios en las llaves primarias no generen inconsistencias.

Por estas razones, consideramos que **ON DELETE CASCADE** y **ON UPDATE CASCADE** son las políticas más adecuadas para nuestro esquema en PostgreSQL.

## Referencias

PostgreSQL Global Development Group. *Documentation: Referential Integrity Constraints*. Disponible en: <https://www.postgresql.org/docs/current/ddl-constraints.html>

W3Schools. *SQL FOREIGN KEY Constraint*. Disponible en: [https://www.w3schools.com/sql/sql\\_foreignkey.asp](https://www.w3schools.com/sql/sql_foreignkey.asp)

DBA Stack Exchange. *Disadvantages of ON DELETE CASCADE*. Disponible en: <https://dba.stackexchange.com/questions/64710/disadvantages-of-on-delete-cascade>

TechTarget. *Foreign key constraints in PostgreSQL*. Disponible en: <https://www.techtarget.com/searchdatamanagement/definition/foreign-key>