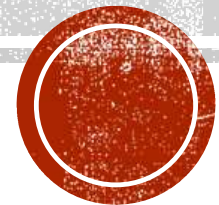


# **REGLAS DE INTEGRIDAD EN EL MODELO RELACIONAL**



**Gestión de Bases de Datos – 1º ASIR**  
**IES Trafalgar**

# REGLA DE INTEGRIDAD DE UNICIDAD DE LA CLAVE PRIMARIA

La **regla de integridad de unicidad** está relacionada con la definición de clave primaria. Concretamente, establece que toda clave primaria que se elija para una relación no deben tener valores repetidos.

Ejemplo

Tenemos la siguiente relación:

SALON		
ID_SALON	ID_EDIFICIO	DIMENSIÓN
101	13	10
102	13	14
103	14	20
103	15	13



En esta relación, dado que la clave primaria está formada por ID-SALON y ID-EDIFICIO, no hay ningún salón que repita tanto id-edificio como número de otro salón. Sin embargo, sí se repiten valores de id-edificio (por ejemplo, 13); y también se repiten valores de id-salón (103). A pesar de ello, el edificio y el número no se repiten nunca al mismo tiempo.



# REGLA DE INTEGRIDAD DE CLAVE PRIMARIA

La **regla de integridad de entidad de la clave primaria** dispone que los atributos de la clave primaria de una relación no puedan tener valores nulos.

Ejemplo

Tenemos la siguiente relación:

SALON		
ID-SALON	ID-EDIFICIO	DIMENSIÓN
101	13	10
102	13	14
103	14	20
103	15	13

Cuando se crean las tablas, los campos que se requieren que sean NO NULOS deben tener la sentencia NOT NULL después de especificar el tipo de dato:

```
Create table salon(  
  Id_salon integer NOT NULL,  
  Id_edificio integer NOT NULL,  
  Dimension integer NOT NULL,  
  Primary key(id_salon,id_edificio));
```



En esta relación, puesto que la clave primaria está formada por ID-EDIFICIO y ID-SALÓN, no hay ningún salón que tenga un valor nulo para id-edificio, ni tampoco para id-salón.

Esta regla es necesaria para que los valores de las claves primarias puedan identificar las tuplas individuales de las relaciones. Si las claves primarias tuviesen valores nulos, es posible que algunas tuplas no se pudieran distinguir.



# REGLA DE INTEGRIDAD REFERENCIAL

La **regla de integridad referencial** está relacionada con el concepto de clave foránea. Concretamente, determina que todos los valores que toma una clave foránea deben ser nulos o valores que existen en la clave primaria que referencia.

Ejemplo

Si tenemos las siguientes relaciones:

SALON		
ID-SALON	ID-EDIFICIO	DIMENSIÓN
101	13	10
102	13	14
103	14	20
103	15	13

ESTUDIANTES				
CODIGO	NOMBRE	APELLIDO	SALON_IDSALON	SALON_IDEDIFICIO
12376876	JUAN	ROJAS	101	13
32345412	FEDERICO	PEREZ	101	13
22312376	ANTONIO	DIAZ	102	13
34345645	FERNANDO	LALINDE	103	15



Donde SALON-ID-EDIFICIO y SALON-ID-SALON de la relación ESTUDIANTES forman una clave foránea que referencia la relación SALÓN. Debe ocurrir que los valores no nulos de SALON-ID-EDIFICIO y SALON-ID-SALON de la relación ESTUDIANTES estén en la relación SALÓN como valores de ID-SALON y ID-EDIFICIO. Por ejemplo, el empleado <22312376, ANTONIO DIAZ, 102,13> tiene el valor 13 para ID-EDIFICIO y el valor 102 para ID-SALÓN, de modo que en la relación SALÓN hay un salón con valor 13 para ID-EDIFICIO y con valor 102 para ID-SALÓN.





# REGLA DE INTEGRIDAD REFERENCIAL

SALON			ESTUDIANTES				
ID-SALON	ID-EDIFICIO	DIMENSIÓN	CODIGO	NOMBRE	APELLIDO	SALON-IDSALON	SALON- ID-EDIFICIO
101	13	10	12376876	JUAN	ROJAS	101	13
102	13	14	32345412	FEDERICO	PEREZ	101	13
103	14	20	22312376	ANTONIO	DIAZ	102	13
103	15	13	34345645	FERNANDO	LALINDE	103	15

Las siguientes operaciones provocarían el incumplimiento de la regla de integridad referencial:

- Inserción de <12764411, JOSE, RIOS, 105,15> en ESTUDIANTES.

💡 | Incumple la regla de integridad referencial porque el salón 105, edificio 15 no existe en la relación SALON.

- Modificación de <34345645, FERNANDO, LALINDE, 103,15,> de ESTUDIANTES por <34345645, FERNANDO, LALINDE, 101,16>.

💡 | Incumple la regla de integridad referencial porque el salón 101, edificio 16 no existe en la relación SALON.



# RESTRICCIONES



# RESTRICT

Es el comportamiento por defecto, que impide realizar modificaciones que atentan contra la integridad referencial.

**DELETE FROM ciudades**  
**WHERE ciudad\_id=1;**

**Error:** Cannot delete or update a parent row: a foreign key constraint fails.

Porque el **DELETE** viola la restricción. Si la fila 1 de ciudades se borrara, las filas 1 y 4 de personas quedarían huérfanas, o sea, sin relación en la tabla ciudades.

personas		
persona_id	persona_nom	ciudad_id
1	Pedro	1
2	Santiago	2
3	Juan	3
4	Andrés	1

\* Si intentamos borrar una fila en ciudades que tenga valores en personas tendremos el error  
**foreign key constraint fails**

\* Si no hay registro relacionado se borrará

**RESTRICT**

ciudades	
ciudad_id	ciudad_nom
1	Galilea
2	Betsaida
3	Patmos
4	<del>Jerusalén</del>

# CASCADE

Borra los registros de la tabla dependiente cuando se borra el registro de la tabla principal (**DELETE**), o actualiza el valor de la clave secundaria cuando se actualiza el valor de la clave referenciada (**UPDATE**).

**DELETE FROM ciudades**  
**WHERE ciudad\_id=1;**

personas		
persona_id	persona_nom	ciudad_id
1	David	1
2	Santiago	2
3	Juan	3
4		1

\* Si intentamos borrar una fila en **ciudades** que tenga valores en **personas** todos los registros relacionados en **personas** se borrarán

CASCADE

ciudades	
ciudad_id	ciudad_nom
1	Jerusalén
2	Betsaida
3	Patmos
4	Jerusalén



# SET NULL

Establece a **NULL** el valor de la clave secundaria cuando se elimina el registro en la tabla principal o se modifica el valor del campo referenciado.

**DELETE FROM ciudades**  
**WHERE ciudad\_id=1;**

Aquí la columna ciudad\_id de la tabla personas establecerá los valores a NULL, en todas las filas cuyo ciudad\_id sea igual a 1. Y como es evidente, se borrará en ciudades la ciudad con id 1.

personas		
persona_id	persona_nom	ciudad_id
1	Pedro	NULL
2	Santiago	2
3	Juan	3
4	Andrés	NULL

\* Si intentamos borrar una fila en **ciudades** que tenga valores en **personas** todos los registros relacionados en **personas** tendrán el valor **NULL** en la columna que es llave foránea

SET NULL

ciudades	
ciudad_id	ciudad_nom
<del>1</del>	<del>Jerusalén</del>
2	Betsaida
3	Patmos
4	Jerusalén

[HTTPS://REXTESTER.COM/ENCX33603](https://rextester.com/encx33603)



# VISUAL CODE

- `sudo snap install code --classic`

