



[Inicio](#) > [SQL](#) > [Lecciones SQL](#) >

## P06 Opciones de integridad referencial

### CONTENIDOS

#### 1 Las opciones para mantener la integridad referencial

##### 1.1 On delete

##### 1.2 On update

## Las opciones para mantener la integridad referencial

Ya se ha practicado con la integridad referencial en sesiones anteriores y se ha visto que el intento de borrar ciertas filas es rechazado por el SGBD si éstas están siendo referenciadas por alguna clave ajena. El sistema está velando por la integridad de los datos almacenados.

No obstante, es posible automatizar y prever estas situaciones expresando en el esquema de la base de datos nuestra voluntad de propagar las operaciones de borrado de filas y de actualización de valores de clave primaria hasta donde haga falta. Antes de mostrar estas opciones, el siguiente ejemplo nos recuerda las restricciones que impone el exigir integridad referencial a las claves ajenas.

```
-- selecciona tu base de datos
-- use zXXX

select '*****          Preparando la BD MiEjemplo...' acción
from dual;
drop table if exists imparte;
drop table if exists asignaturas;
drop table if exists profesores;
```

```
create table profesores
( dni varchar(10), nombre varchar(40), categoria char(4), ingreso
date, primary key (dni)) engine=innodb;
```

```
create table asignaturas
( codigo char(5), descripcion varchar(35), creditos decimal(3,1),
creditosp decimal(3,1),
primary key (codigo)) engine=innodb;
```

```
insert into profesores (select * from ejemplo.profesores);
insert into asignaturas (select * from ejemplo.asignaturas);
```

```
create table imparte ( dni varchar(10), asignatura char(5),
primary key (dni, asignatura),
foreign key (dni) references profesores (dni),
foreign key (asignatura) references asignaturas (codigo))
engine=innodb;
```

```
insert into imparte (select * from ejemplo.imparte);
```

```
select * from asignaturas;
select * from profesores;
select * from imparte;
```

\*\*\*\*\* estado de la BD INICIAL

codigo	descripcion	creditos	creditosp
DGBD	DISEÑO Y GESTION DE BASES DE DATOS	6.0	3.0
<b>FBD</b>	<b>FUNDAMENTOS DE LAS BASES DE DATOS</b>	<b>6.0</b>	<b>1.5</b>
FP	FUNDAMENTOS DE LA PROGRAMACION	9.0	4.5
<b>HI</b>	<b>HISTORIA DE LA INFORMATICA</b>	<b>4.5</b>	
PC	PROGRAMACION CONCURRENTES	6.0	1.5

dni	nombre	categoria	ingreso
21111222	EVA GOMEZ	TEU	1993-10-01
21222333	MANUEL PALOMAR	TEU	1989-06-16
21333444	RAFAEL ROMERO	ASO6	1992-06-16

dni	asignatura
21111222	DGBD
21111222	FBD

21333444	PC
----------	----

```
delete from asignaturas where codigo = 'FBD';
```

La ejecución fallará porque hay profesores que imparten la asignatura FBD.

```
delete from asignaturas where codigo = 'HI';
```

Se cumple la orden porque nadie imparte HI.

```
select * from asignaturas;
select * from profesores;
select * from imparte;;
```

\*\*\*\*\* estado de la BD FINAL

codigo	descripcion	creditos	creditosp
DGBD	DISEÑO Y GESTION DE BASES DE DATOS	6.0	3.0
<b>FBD</b>	<b>FUNDAMENTOS DE LAS BASES DE DATOS</b>	<b>6.0</b>	<b>1.5</b>
FP	FUNDAMENTOS DE LA PROGRAMACION	9.0	4.5
PC	PROGRAMACION CONCURRENTES	6.0	1.5

dni	nombre	categoria	ingreso
21111222	EVA GOMEZ	TEU	1993-10-01
21222333	MANUEL PALOMAR	TEU	1989-06-16
21333444	RAFAEL ROMERO	ASO6	1992-06-16

dni	asignatura
21111222	DGBD
21111222	FBD
21333444	PC

## On delete

## Propagar

En ciertos sistemas de información es posible redefinir las restricciones de clave ajena para que no se den estos mensajes de error. Ello es posible mediante la cláusula ON DELETE al crear una tabla:

```
FOREIGN KEY (columna[,columna[, ...]]) REFERENCES tabla (clave
primaria)
ON DELETE {CASCADE | SET NULL}
```

La acción a realizar ante el borrado de una fila que está siendo referenciada por alguna clave ajena puede ser el propagar la operación (ON DELETE CASCADE) o anular (ON DELETE SET NULL), dependiendo de la decisión del diseñador de la base de datos.

Veamos ahora el efecto de utilizar la opción ON DELETE CASCADE.

```
drop table if exists imparte;
delete from asignaturas;
delete from profesores;
insert into asignaturas (select * from ejemplo.asignaturas);
insert into profesores(select * from ejemplo.profesores);

create table imparte( dni varchar(10), asignatura char(5),
primary key (dni, asignatura),
foreign key (dni) references profesores(dni),
foreign key (asignatura) references asignaturas(codigo)
ON DELETE CASCADE
) engine=innodb;

insert into imparte (select * from ejemplo.imparte);

select * from asignaturas;
select * from profesores;
select * from imparte;
```

codigo	descripcion	creditos	creditosp
DGBD	DISEÑO Y GESTION DE BASES DE DATOS	6.0	3.0
<b>FBD</b>	<b>FUNDAMENTOS DE LAS BASES DE DATOS</b>	<b>6.0</b>	<b>1.5</b>
FP	FUNDAMENTOS DE LA PROGRAMACION	9.0	4.5
<b>HI</b>	<b>HISTORIA DE LA INFORMATICA</b>	<b>4.5</b>	

PC	PROGRAMACION CONCURRENTE	6.0	1.5
----	--------------------------	-----	-----

dni	nombre	categoria	ingreso
21111222	EVA GOMEZ	TEU	1993-10-01
21222333	MANUEL PALOMAR	TEU	1989-06-16
21333444	RAFAEL ROMERO	ASO6	1992-06-16

dni	asignatura
21111222	DGBD
21111222	FBD
21333444	PC

El borrado de FBD provoca borrados automáticos en IMPARTE y el borrado de HI sigue sin generar problemas

```
delete from asignaturas where codigo = 'FBD';
delete from asignaturas where codigo = 'HI';
```

codigo	descripcion	creditos	creditosp
DGBD	DISEÑO Y GESTION DE BASES DE DATOS	6.0	3.0
FP	FUNDAMENTOS DE LA PROGRAMACION	9.0	4.5
PC	PROGRAMACION CONCURRENTE	6.0	1.5

dni	nombre	categoria	ingreso
21111222	EVA GOMEZ	TEU	1993-10-01
21222333	MANUEL PALOMAR	TEU	1989-06-16
21333444	RAFAEL ROMERO	ASO6	1992-06-16

dni	asignatura
21111222	DGBD
21333444	PC

Sin embargo, en la CAj a PROFESORES no hemos definido método alguno para mantener la IR.

```
delete from profesores where dni = '21111222';
```

Falla porque este profesor imparte DGBD

```
delete from profesores where dni = '21222333';
```

Éxito ya que no imparte ninguna asignatura

codigo	descripcion	creditos	creditosp
DGBD	DISEÑO Y GESTION DE BASES DE DATOS	6.0	3.0
FP	FUNDAMENTOS DE LA PROGRAMACION	9.0	4.5
PC	PROGRAMACION CONCURRENTES	6.0	1.5

dni	nombre	categoria	ingreso
21111222	EVA GOMEZ	TEU	1993-10-01
21333444	RAFAEL ROMERO	ASO6	1992-06-16

dni	asignatura
21111222	DGBD
21333444	PC

## Anular

Si la modificación de una clave ajena es ON DELETE SET NULL, la acción que llevará a cabo automáticamente el SGBD es la de poner NULOS en aquellos casos en que la integridad referencial se vea comprometida.

Esta definición tiene más dificultad de aplicación puesto que prevalecen las definiciones de VALOR NO NULO. Por ejemplo, es inútil utilizarla en IMPARTE.DNI ya que forma parte de la clave primaria y no admite nulos en ningún caso. Vamos a cambiar el esquema de IMPARTE.

```
drop table if exists imparte;
delete from asignaturas;
delete from profesores;
insert into profesores (select * from ejemplo.profesores);
insert into asignaturas (select * from ejemplo.asignaturas);

create table imparte (
  ficha integer,
  dni varchar(10), asignatura char(5),
  primary key (ficha),
  foreign key (dni) references profesores (dni) ON DELETE SET NULL,
  foreign key (asignatura) references asignaturas (codigo) )
engine=innodb;
```

Las claves ajenas, ahora, permiten nulos.

```
insert into imparte values (1,'21111222','FBD');
insert into imparte values (2,'21111222','DGBD');
insert into imparte values (3,'21333444','PC');
```

```
select * from asignaturas;
select * from profesores;
select * from imparte;
```

codigo	descripcion	creditos	creditosp
DGBD	DISEÑO Y GESTION DE BASES DE DATOS	6.0	3.0
FBD	FUNDAMENTOS DE LAS BASES DE DATOS	6.0	1.5
FP	FUNDAMENTOS DE LA PROGRAMACION	9.0	4.5
HI	HISTORIA DE LA INFORMATICA	4.5	
PC	PROGRAMACION CONCURRENTES	6.0	1.5

dni	nombre	categoria	ingreso
21111222	EVA GOMEZ	TEU	1993-10-01
21222333	MANUEL PALOMAR	TEU	1989-06-16
21333444	RAFAEL ROMERO	ASO6	1992-06-16

ficha	dni	asignatura
1	21111222	DGBD
2	21111222	FBD
3	21333444	PC

```
delete from profesores where dni = '21111222';
```

codigo	descripcion	creditos	creditosp
DGBD	DISEÑO Y GESTION DE BASES DE DATOS	6.0	3.0
FBD	FUNDAMENTOS DE LAS BASES DE DATOS	6.0	1.5
FP	FUNDAMENTOS DE LA PROGRAMACION	9.0	4.5
HI	HISTORIA DE LA INFORMATICA	4.5	
PC	PROGRAMACION CONCURRENTES	6.0	1.5

dni	nombre	categoria	ingreso
21222333	MANUEL PALOMAR	TEU	1989-06-16
21333444	RAFAEL ROMERO	ASO6	1992-06-16

ficha	dni	asignatura
1		DGBD
2		FBD

3	21333444	PC
---	----------	----

## On update

La definición ON UPDATE se rige por los mismos parámetros que la anterior, se puede elegir entre propagar o anular (o rechazar, si no definimos nada).

Nótese que se pueden hacer todas las combinaciones posibles en todas las claves ajenas, no necesariamente ha de ser todas propagar o todas anular, ni tampoco tienen por qué coincidir en la misma clave ajena.

```
drop table if exists imparte;
delete from asignaturas;
delete from profesores;
insert into asignaturas (select * from ejemplo.asignaturas);
insert into profesores(select * from ejemplo.profesores);

create table imparte( dni varchar(10), asignatura char(5),
primary key (dni, asignatura),
foreign key (dni) references profesores(dni)
ON DELETE CASCADE ON UPDATE CASCADE ,
foreign key (asignatura) references asignaturas(codigo)
ON DELETE CASCADE ON UPDATE CASCADE
) engine=innodb;

insert into imparte (select * from ejemplo.imparte);

select * from asignaturas;
select * from profesores;
select * from imparte;
```

codigo	descripcion	creditos	creditosp
DGBD	DISEÑO Y GESTION DE BASES DE DATOS	6.0	3.0
<b>FBD</b>	<b>FUNDAMENTOS DE LAS BASES DE DATOS</b>	<b>6.0</b>	<b>1.5</b>
FP	FUNDAMENTOS DE LA PROGRAMACION	9.0	4.5
HI	HISTORIA DE LA INFORMATICA	4.5	
PC	PROGRAMACION CONCURRENTES	6.0	1.5



dni	nombre	categoria	ingreso
<b>21111222</b>	<b>EVA GOMEZ</b>	<b>TEU</b>	<b>1993-10-01</b>
21222333	MANUEL PALOMAR	TEU	1989-06-16
21333444	RAFAEL ROMERO	ASO6	1992-06-16

dni	asignatura
<b>21111222</b>	DGBD
<b>21111222</b>	<b>FBD</b>
21333444	PC

```
update asignaturas set codigo = 'AAA' where codigo = 'FBD';
update profesores set dni = '33' where dni = '21111222';
```

codigo	descripcion	creditos	creditosp
DGBD	DISEÑO Y GESTION DE BASES DE DATOS	6.0	3.0
<b>AAA</b>	<b>FUNDAMENTOS DE LAS BASES DE DATOS</b>	<b>6.0</b>	<b>1.5</b>
FP	FUNDAMENTOS DE LA PROGRAMACION	9.0	4.5
HI	HISTORIA DE LA INFORMATICA	4.5	
PC	PROGRAMACION CONCURRENTES	6.0	1.5

dni	nombre	categoria	ingreso
<b>33</b>	<b>EVA GOMEZ</b>	<b>TEU</b>	<b>1993-10-01</b>
21222333	MANUEL PALOMAR	TEU	1989-06-16
21333444	RAFAEL ROMERO	ASO6	1992-06-16

dni	asignatura
<b>33</b>	DGBD
<b>33</b>	<b>AAA</b>
21333444	PC

Obviamente, se puede establecer cualquier combinación de métodos que queramos utilizar, como por ejemplo

```
create table imparte( dni varchar(10), asignatura char(5),
primary key (dni, asignatura),
foreign key (dni) references profesores(dni)
ON UPDATE CASCADE ,
foreign key (asignatura) references asignaturas(codigo)
```

```
ON DELETE SET NULL ON UPDATE CASCADE
```

```
) engine=innodb;
```

En el ejemplo anterior se utilizan RECHAZAR, PROPAGAR Y ANULAR entre las dos claves ajenas de la tabla.



FBDdocs por BDgite se encuentra bajo una [Licencia Creative Commons Atribución-CompartirIgual 3.0 Unported](http://creativecommons.org/licenses/by-sa/3.0/). Basada en una obra en <http://fbddocs.dlsi.ua.es>. Permisos que vayan más allá de lo cubierto por esta licencia pueden encontrarse en <http://fbddocs.dlsi.ua.es/autores>.

BDgite (GITE-11014-UA),  
Departamento de Lenguajes y Sistemas Informáticos,  
Universidad de Alicante

[Iniciar sesión](#) | [Informar de uso inadecuado](#) | [Imprimir página](#) | Con la tecnología de [Google Sites](#)