

MySQL

DOCS_

Cómo crear y manejar
constraints en MySQL



Cómo crear y manejar constraints en MySQL

Los constraint son validaciones que podemos aplicar dentro de nuestra base de datos. Estas validaciones se basan por supuesto, en las reglas de negocios que tengamos definidas para nuestra base de datos. Las reglas que definamos se almacenan dentro de cada tabla y pueden ayudar a limitar el tipo de datos o preservar la integridad de la base de datos

Sintaxis para un constraint

La sintaxis para crear un constraint es muy sencilla:

```
1 CONSTRAINT Nombre_Constraint CHECK(Condiciones)
```

Siendo:

- **Nombre_Constraint:** El nombre que podemos definir para el constraint
- **Condiciones:** La condición o condiciones que necesitamos que se cumplan, como WHERE acepta la mayoría de conectores (AND, OR, NOT, BETWEEN) y operadores lógicos (=, <, >, <=, >=).

Creando nuestro primer constraint

Comencemos definiendo una tabla con un constraint. En un primer ejemplo, crearemos una tabla de contactos que sólo permita insertar a personas mayores de edad. Observemos:

```
CREATE TABLE CONTACTOS(  
  conId int not null auto_increment,  
  conNombre varchar(30) not null,  
  conEdad int not null,  
  conCiudad varchar(30),  
  primary key (conId),  
  CONSTRAINT ChkEdad CHECK (conEdad>18)  
);
```

Intentemos insertar ahora un primer registro.

```
insert into contactos values(NULL, "Francisco López",22,"Veracruz");
```

Como podemos observar, el dato ha sido insertado con éxito.

Intentemos insertar un segundo registro.

```
insert into contactos values(NULL, "Maria Peña",15,"Veracruz");
```

Dado que no se cumple la condición, este será rechazado.

Añadiendo otro constraint con **ALTER TABLE**

Supongamos que necesitamos limitar las ciudades, podemos limitar eso con un constraint (o con un campo tipo **SET**), para añadir un nuevo constraint a nuestra tabla, hemos de usar **ALTER TABLE**. Por ejemplo:

```
ALTER TABLE contactos ADD CONSTRAINT chkCiudad CHECK  
(conCiudad="Veracruz" or ConCiudad="Xalapa");
```

Importante notar el uso del conector lógico **OR** para limitar las opciones a escoger. Entonces, el siguiente contacto será aceptado.

```
insert into contactos values(NULL, "Saúl Feria",33,"Veracruz");
```

Definitivamente cumple con los dos constraint establecidos.

Pero el siguiente contacto no será aceptable.

```
insert into contactos values(NULL, "Silvia López",12,"Orizaba");
```

Definitivamente no cumple ninguno de los constraint.

¿Cómo puedo ver los constraint de una tabla?

Suele pasar que olvidamos a veces los constraint que hemos definido, así que para verlos, solo necesitamos escribir el comando **SHOW CREATE TABLE** seguido del nombre de nuestra tabla. Por ejemplo:

```
show create table contactos;
```

Nos mostrará lo siguiente:

Eliminar un constraint de una tabla

Para eliminar un constraint, utilizamos **ALTER TABLE** junto a **DROP CONSTRAINT** y le pasamos el nombre de nuestro constraint como elemento final. Observemos el siguiente comando para eliminar la regla que colocamos sobre la ciudad.

```
ALTER TABLE contactos DROP CONSTRAINT chkCiudad;
```

Solo tenemos que recordar el nombre que le dimos.

Añadir una llave foránea con un constraint

Hagamos una tabla de citas.

```
CREATE TABLE citas(  
  citContacto int not null,  
  citFecha date not null,  
  citHora time not null,  
  citAsunto varchar(40)  
);
```

Si olvidamos añadir la llave foránea, podemos hacer con un constraint. La sintaxis es algo similar a la creación de un constraint simple, pero sin la sentencia **CHECK**. Por supuesto, utilizaremos **ALTER TABLE**.

```
ALTER TABLE citas ADD CONSTRAINT fkContacto FOREIGN  
KEY(citContacto) REFERENCES contactos(conId) on update cascade;
```

Como podemos observar, la tabla ha aceptado la llave foránea.

Entonces, la siguiente cita será aceptada.

```
insert into citas values(2,"2021-10-20","11:30:00","Comer juntos");
```

Ya que el contacto número dos existe en la tabla correspondiente.

La siguiente cita no será aceptada.

```
insert into citas values(22,"2021-10-20","11:30:00","Ir al parque");
```

Ya que el contacto número 22, todavía no existe.