

11 - Valores null (is null)

"null" significa "dato desconocido" o "valor inexistente". No es lo mismo que un valor "0", una cadena vacía o una cadena literal "null".

A veces, puede desconocerse o no existir el dato correspondiente a algún campo de un registro. En estos casos decimos que el campo puede contener valores nulos.

Por ejemplo, en nuestra tabla de libros, podemos tener valores nulos en el campo "precio" porque es posible que para algunos libros no le hayamos establecido el precio para la venta.

En contraposición, tenemos campos que no pueden estar vacíos jamás.

Veamos un ejemplo. Tenemos nuestra tabla "libros". El campo "titulo" no debería estar vacío nunca, igualmente el campo "autor". Para ello, al crear la tabla, debemos especificar que dichos campos no admitan valores nulos:

```
create table libros(  
  titulo varchar(30) not null,  
  autor varchar(20) not null,  
  editorial varchar(15) null,  
  precio float  
);
```

Para especificar que un campo no admita valores nulos, debemos colocar "not null" luego de la definición del campo.

En el ejemplo anterior, los campos "editorial" y "precio" si admiten valores nulos.

Cuando colocamos "null" estamos diciendo que admite valores nulos (caso del campo "editorial"); por defecto, es decir, si no lo aclaramos, los campos permiten valores nulos (caso del campo "precio").

Si ingresamos los datos de un libro, para el cual aún no hemos definido el precio podemos colocar "null" para mostrar que no tiene precio:

```
insert into libros (titulo,autor,editorial,precio)  
values('El aleph','Borges','Emece',null);
```

Note que el valor "null" no es una cadena de caracteres, no se coloca entre comillas.

Entonces, si un campo acepta valores nulos, podemos ingresar "null" cuando no conocemos el valor.

También podemos colocar "null" en el campo "editorial" si desconocemos el nombre de la editorial a la cual pertenece el libro que vamos a ingresar:

```
insert into libros (titulo,autor,editorial,precio)  
values('Alicia en el pais','Lewis Carroll',null,25);
```

Si intentamos ingresar el valor "null" en campos que no admiten valores nulos (como "titulo" o "autor"), SQL Server no lo permite, muestra un mensaje y la inserción no se realiza; por ejemplo:

```
insert into libros (titulo,autor,editorial,precio)  
values(null,'Borges','Siglo XXI',25);
```

Para ver cuáles campos admiten valores nulos y cuáles no, podemos emplear el procedimiento almacenado "sp_columns" junto al nombre de la tabla. Nos muestra mucha información, en la columna "IS_NULLABLE" vemos que muestra "NO" en los campos que no permiten valores nulos y "YES" en los campos que si los permiten.

Para recuperar los registros que contengan el valor "null" en algún campo, no podemos utilizar los operadores relacionales vistos anteriormente: = (igual) y <> (distinto); debemos utilizar los operadores "is null" (es igual a null) y "is not null" (no es null):

```
select * from libros
where precio is null;
```

La sentencia anterior tendrá una salida diferente a la siguiente:

```
select * from libros
where precio=0;
```

Con la primera sentencia veremos los libros cuyo precio es igual a "null" (desconocido); con la segunda, los libros cuyo precio es 0.

Igualmente para campos de tipo cadena, las siguientes sentencias "select" no retornan los mismos registros:

```
select * from libros where editorial is null;
select * from libros where editorial='';
```

Con la primera sentencia veremos los libros cuya editorial es igual a "null", con la segunda, los libros cuya editorial guarda una cadena vacía.

Entonces, para que un campo no permita valores nulos debemos especificarlo luego de definir el campo, agregando "not null". Por defecto, los campos permiten valores nulos, pero podemos especificarlo igualmente agregando "null".

Servidor de SQL Server instalado en forma local.

Ingresemos el siguiente lote de comandos en el SQL Server Management Studio:

```
if object_id('libros') is not null
    drop table libros;

create table libros(
    titulo varchar(30) not null,
    autor varchar(30) not null,
    editorial varchar(15) null,
    precio float
);

go

-- Agregamos un registro a la tabla con valor nulo para el campo "precio":
insert into libros (titulo,autor,editorial,precio)
values('El aleph','Borges','Emece',null);

-- Ingresamos otro registro, con valor nulo para el campo "editorial"
insert into libros (titulo,autor,editorial,precio)
values('Alicia en el pais','Lewis Carroll',null,0);
```

```

-- Veamos lo que sucede si intentamos ingresar el valor "null"
-- en campos que no lo admiten, como "titulo":
insert into libros (titulo,autor,editorial,precio)
  values(null,'Borges','Siglo XXI',25);

exec sp_columns libros;

-- Dijimos que el valor "null" no es lo mismo que una cadena vacía.
-- Vamos a ingresar un registro con cadena vacía para el campo "editorial":
insert into libros (titulo,autor,editorial,precio)
  values('Uno','Richard Bach','',18.50);

-- Ingresamos otro registro, ahora cargamos una cadena vacía en el campo "titulo":
insert into libros (titulo,autor,editorial,precio)
  values('','Richard Bach','Planeta',22);

select * from libros;

-- Recuperemos los registros que contengan el valor "null" en el campo "precio":
select * from libros
  where precio is null;

-- La sentencia anterior tendrá una salida diferente a la siguiente:
select * from libros
  where precio=0;

-- Recuperemos los libros cuyo nombre de editorial es "null":
select * from libros
  where editorial is null;

-- Ahora veamos los libros cuya editorial almacena una cadena vacía:
select * from libros
  where editorial='';

-- Para recuperar los libros cuyo precio no sea nulo tipeamos:
select * from libros
  where precio is not null;

```

Tenemos como resultado:

```
create table libros(  
    titulo varchar(30) not null,  
    autor varchar(30) not null,  
    editorial varchar(15) null,  
    precio float  
);  
  
-- Agregamos un registro a la tabla con valor nulo para el campo "precio":  
insert into libros (titulo,autor,editorial,precio)  
values('El aleph','Borges','Emece',null);  
  
-- Ingresamos otro registro, con valor nulo para el campo "editorial"  
insert into libros (titulo,autor,editorial,precio)  
values('Alicia en el pais','Lewis Carroll',null,0);  
  
-- Veamos lo que sucede si intentamos ingresar el valor "null" en campos que no lo admiten, como "titulo"  
insert into libros (titulo,autor,editorial,precio)  
values(null,'Borges','Siglo XXI',25);
```

100 %

Results Messages

(1 row affected)

(1 row affected)

Msg 815, Level 15, State 2, Line 20

No se puede insertar el valor NULL en la columna 'titulo', tabla 'bdi.dbo.libros'. La columna no admite ya.
Se terminó la instrucción.

(4 rows affected)

(1 row affected)

(1 row affected)

(4 rows affected)

(1 row affected)

100 %

Query completed with errors.

DIEGO-PC (14.0 RTM) | DIEGO-PC