

U.T. VII.- Inserción de datos en MySQL.

Ejercicio 1.

Crear una BD con las siguientes tablas:

NULOS

Campo	Tipo	Restricción
IntNulo	int	
CharNulo	char	

NONULOS

Campo	Tipo	Restricción
IntNoNulo	int	NOT NULL
CharNoNulo	char	NOT NULL

CLAVENULOS

Campo	Tipo	Restricción
IntNulo	int	PRIMARY
CharNulo	char	KEY

CLAVENONULOS

Campo	Tipo	Restricciones	
Int NoNulo	int	NOT NULL	PRIMARY
CharNoNulo	char	NOT NULL	KEY

- Al ejecutar cada una de las siguientes sentencias ¿qué debería ocurrir según el estándar relacional?
- ¿Qué ocurre en MySQL?
- Repetirlo para el resto de tablas.

```
insert Nulos values(1,'a');
select * from Nulos;
insert Nulos values(2,NULL);
/* insert Nulos(IntNulos) values(2)*/
select * from Nulos;
insert Nulos values(NULL,'b');
/*insert Nulos(CharNulos) values('b');*/
select * from Nulos;
insert Nulos values(NULL,NULL);
/* insert Nulos values();*/
select * from Nulos;
insert Nulos values(NULL,NULL); )
/* insert Nulos values();*/
select * from Nulos;
```

- Este comportamiento es debido a que, cuando en una sentencia **INSERT** no se especifica un valor para una más columnas de la tabla, MySQL les asigna un valor por defecto.
 - o El 0 para campos numéricos, un espacio en blanco para datos de tipo carácter, 0000-00-00 para datos de tipo fecha, etc.
 - o Podemos especificar el valor por defecto que debe tomar una columna mediante la cláusula **DEFAULT default_value** de la sentencia **CREATE TABLE** .
- Podemos modificar esta forma "*no relacional*" de comportamiento cambiando el "*modo*" en el que se ejecuta SQL.

Manual: **13.2.4. Sintaxis de INSERT.**

Ejercicio 2.

- Borrar la BD creada en el ejercicio anterior.
- Cambiar el modo de funcionamiento de SQL a **STRICT_TRANS_TABLES**.
- Volver a ejecutar las sentencias **INSERT** del ejercicio anterior y observad cómo se comportan.