

# Inserción de datos

# Insertar registros en una tabla

Para insertar datos en una tabla utilizamos la sentencia **INSERT**. Con ella, podemos **añadir registros uno a uno o añadir tantos registros como deseemos en una sola sentencia**.

Existen distintas formas de ingresar registros en una tabla. Dichas formas son reconocidas con los siguientes nombres:

- Manera completa.
- Manera SQL.
- Manera simplificada.

INSERT



## Manera completa

Sintaxis:

```
INSERT INTO PRODUCTOS (Nombre, Precio, Marca, Categoria, Stock, Disponible)  
VALUES ('iPhone 5', 499.99, 'Apple', 'Smartphone', 500, false);
```

En este caso, a continuación del nombre de la tabla, se deben **especificar los campos que ella contiene**. Posteriormente, se especifican **los valores** a cargar en cada uno de ellos.

## Manera SQL (solo soportado por el motor MySQL/MariaDB)

Sintaxis:

```
INSERT INTO PRODUCTOS  
SET Nombre = 'iPhone 5', Precio = 499.99, Marca = 'Apple', Categoria =  
'Smartphone', Stock = 500, Disponible = false;
```

En este caso, **los datos a cargar o insertar en cada uno de los campos se definen junto a su nombre**. Es solo una manera distinta de insertar el registro en la tabla *PRODUCTOS*.

## Manera simplificada

Sintaxis:

```
INSERT INTO Productos  
VALUES ('iPhone 5', 499.99, 'Apple', 'Smartphone', 500, false)
```

En este caso, **solo se especifican los valores** a cargar en cada uno de los campos. Se la denomina *manera simplificada*, dado que **no se especifican los nombres de los campos** luego del nombre de la tabla.

Solo se detallan los datos a cargar en cada campo de la tabla. Es muy importante respetar el orden: **los datos se deben especificar en el mismo orden en que figuran las columnas en la tabla.**

# Valores nulos: *NULL*

La expresión *NULL* significa "**dato desconocido**" o "**valor inexistente**". No es lo mismo que un valor 0 en un campo numérico, una cadena vacía o una cadena de texto con la palabra *NULL* en un campo de tipo texto.

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

Por ejemplo, en una tabla con el nombre *Productos*, se puede tener valores nulos en el campo *precio*, en el caso de que, para algunos productos, no se haya establecido el precio para la venta.

En contraposición, una tabla puede contener campos que **no pueden quedar vacíos**, como los que identifican cada registro (códigos de identificación) que son clave primaria.

**Nota: por defecto** (si no lo aclaramos en la creación de la tabla), **los campos permiten valores nulos**.



Ejemplo:

```
INSERT INTO Productos (Nombre, Precio, Marca, Categoria, Presentacion, Stock)
VALUES ('iPhone 7S', NULL, 'Apple', 'Smartphone', '16GB', 500);
```

Aclaraciones:

- Nótese que para el campo *Precio*, el valor **NULL no es una cadena de caracteres**. Por ende, **no se coloca entre comillas**. Si un campo acepta valores nulos, podemos ingresar *NULL* cuando no conocemos el valor.
- Además, si una **columna** fue definida como **no obligatoria** en la sentencia *CREATE TABLE*, **puede no incluirse en el listado de campos** (puede entenderse como que el campo "no es obligatorio" de asignar un valor).

## Consulta de datos anexados: *DML INSERT*

Se pueden **insertar datos** (provenientes de otra tabla) en una tabla a partir de una sentencia ***SELECT***. De este modo, será posible realizar una **inserción masiva de datos desde una tabla hacia otra en una sola instrucción** o sentencia.

Sintaxis:

```
INSERT INTO TablaDestino (Columna1, ..., ColumnaX)
SELECT (Columna1, ..., ColumnaX) FROM TablaOrigen;
```

**Nota:** la tabla de destino (llamada *TablaDestino* en el ejemplo anterior) debe tener la **misma estructura que la consulta *SELECT***. Es decir, la misma cantidad de columnas y tipos de datos compatibles.





**¡Sigamos  
trabajando!**