

Sentencia UPDATE

La sentencia UPDATE permite modificar los datos contenidos en las tablas. Existen varias opciones de sintaxis:

Opción para modificar datos de una única tabla:

```
UPDATE [LOW_PRIORITY] nombre_tabla SET nombre_columna = {expresión |  
DEFAULT | NULL} [, SET nombre_columna = {expresión | DEFAULT | NULL}] [,  
...] [WHERE condición] [ORDER BY expresión] [LIMIT número_filas]
```



Opción para modificar datos de varias tablas:

```
UPDATE [LOW_PRIORITY] referencia_tablas SET nombre_columna = {expresión |  
DEFAULT | NULL} [, nombre_columna = {expresión | DEFAULT | NULL}] [, ...]  
[WHERE condición]
```

La opción LOW_PRIORITY permite indicar el nivel de prioridad que tiene la operación de inserción. Con esta opción, el servidor espera a que no haya cliente leyendo la tabla para realizar la modificación. Solo afecta a tablas con motores que usan bloqueo a nivel de tabla (MyISAM, MEMORY, MERGE).

Cuando se utiliza la opción para actualizar varias tablas, la sentencia UPDATE actualiza las filas de las tablas nombradas en referencia_tablas que cumplen las condiciones establecidas en la cláusula WHERE. Cada juego de filas se actualiza una única vez, aunque se cumplan varias veces las condiciones.

La cláusula SET indica las columnas que se van a modificar y los valores nuevos que van a recibir.

La cláusula WHERE es opcional y especifica las filas que deben actualizarse.

IMPORTANTE: si no se escribe la cláusula WHERE, se actualizan todas las filas indicadas.

La cláusula ORDER BY es opcional e indica el orden en el que se actualizan las filas. No se puede utilizar en la actualización de múltiples tablas.

Ejemplo:

```
UPDATE irpf SET tipo_impuesto=ROUND(tipo_impuesto*1.05,2);
```

Ejemplo:

```
UPDATE libros, pedidos  
SET pedidos.cantidad = pedidos.cantidad + 2,  
    libros.stock = libros.stock - 2  
WHERE  
    libros.id_libro = pedidos.id_libro  
    AND pedidos.id_pedido = 1002;
```