

Bases de Datos

Unidad 6:
Edición simple de los datos.
Parte 2: UPDATE

Actualización de datos. UPDATE

La instrucción para realizar modificaciones de los datos es:

UPDATE

La sintaxis de UPDATE es:

UPDATE [IGNORE] tabla | combinación_de_tablas
SET columna1=expresión1, columna2=expresión2,
WHERE condición;

Tras UPDATE, se escribe la tabla en la que se van a actualizar datos o, en su caso, la combinación de tablas afectadas por la operación.

Modifica los valores de las columnas indicadas en SET con el resultado de las expresiones. Si se usa WHERE, sólo se modifican las filas que cumplen la condición.

Dentro de la sintaxis, incluso se permite usar ORDER BY para establecer el orden en que se van modificando datos de filas.

NOTA :

SET SQL_SAFE_UPDATES = 0; # Esta instrucción desactiva el modo seguro para UPDATES

SET SQL_SAFE_UPDATES = 1; # Esta instrucción reactiva el modo seguro para UPDATES

Condiciones de la sintaxis y ejecución de UPDATE

1.- Detrás de **UPDATE** se puede especificar:

- **El nombre de una tabla** si la modificación sólo afecta a esa tabla y está condicionada al contenido de esa tabla.
- La combinación de varias tablas cuando la modificación afecta a dos o más tablas relacionadas o bien está condicionada al contenido de varias tablas.

2.- Detrás de **SET** se especifican los valores que se van a asignar a las columnas que se quieren modificar. Estos valores pueden ser:

- Valores constantes.
- El resultado de una expresión que utiliza o no los valores almacenados en otras columnas o en la misma columna.
- El resultado de una función.
- El resultado de una subconsulta.

Condiciones de la sintaxis y ejecución de UPDATE

3.- Si para calcular una expresión, se usa una columna que también se modifica, **el valor que se usa es el almacenado antes de hacer la modificación.**

4.- **Es indiferente el orden en el que se especifiquen las columnas a modificar** puesto que las modificaciones se ejecutan calculando primero los nuevos valores que va a haber en una fila y modificando después el contenido de la fila con esos valores.

5.- Las expresiones deben ser **calculables a partir de los valores correspondientes a la fila que se esté modificando.**

6.- Con la cláusula **WHERE** se indica la condición que deben cumplir las filas que se van a modificar. Si no se usa WHERE, la modificación afectará a todas las filas de la tabla.

Condiciones de la sintaxis y ejecución de UPDATE

7.- Al modificar el contenido de una columna, **está columna deberá cumplir todas las condiciones de dominio, clave principal, clave ajena e integridad referencial** que se tengan que cumplir en función de las restricciones establecidas en las tablas de la base de datos.

8.- **Si se modifica una clave principal que tiene relación de integridad referencial con actualización en cascada** con respecto a una clave ajena en otra tabla, el valor de la clave ajena se modifica con el nuevo valor de la clave principal en todas las filas relacionadas.

9.- La **cláusula IGNORE** no aborta el proceso de actualización cuando se trata de actualizar varias filas y algunas no pueden actualizarse por conflictos de clave primaria, clave ajena, errores de conversión de datos, etc. Las que producen errores no se actualizan y si que se actualizan las que no los producen. Si no se usa IGNORE y alguna fila produce error, se rechaza por completo la actualización

Actualización de datos. UPDATE

Ejemplos de uso de UPDATE

Comprueba previamente el contenido del contrato 19.

Ejemplo 1: Modificar el contrato número 19 para que contenga fecha final 14 de enero de 2020 y kilómetros finales 48111.

UPDATE contratos set ffin='2020-01-14', kfin=48111 WHERE numcontrato=19;

Comprueba el contenido del contrato 19 tras ejecutar UPDATE.

!!! Hay que tener mucho cuidado al probar o realizar estas instrucciones. Si en este caso no hubiéramos escrito la condición WHERE, modificaríamos las fechas y kilómetros de todos los contratos existentes.

Actualización de datos. UPDATE

Ejemplos de uso de UPDATE

Ejemplo 2: Modificar la columna alquilado del vehículo 7839JDR para que indique que está disponible.

```
UPDATE automoviles SET alquilado=false WHERE matricula='7839JDR';
```

Ejemplo 3: Modificar la columna alquilado del vehículo 7839JDR para que contenga lo opuesto a lo que contenía.

```
UPDATE automoviles SET alquilado=NOT alquilado WHERE matricula='7839JDR';
```

Ejemplos de uso de UPDATE

Ejemplo 4: Modificar la columna precio de todos los automóviles de precio de alquiler superior a 100 euros para que su precio se reduzca en un 30% y en un 25% el de los de alquiler inferior a 100 euros. El precio debe quedar con dos decimales.

Ojo, dado que hay que reducir precios, es importante reducir primero los de precio inferior. Si no se hace así, puede que reduzcamos el precio dos veces para algún coche.

```
UPDATE automoviles SET precio=round(precio*0.75,2) WHERE precio<100;  
UPDATE automoviles SET precio=round(precio*0.7,2) WHERE precio>=100;
```

Se podría dar una solución con una sola instrucción usando la función IF:

```
UPDATE automoviles  
SET precio=if(precio<100,round(precio*0.75,2), round(precio*0.7,2));
```


Ejemplos de uso de UPDATE

Antes de realizar la siguiente instrucción comprueba cuales son los contratos realizados para el coche de matrícula 3273JGH:

Ejemplo 5: Modificar la matrícula del automóvil de matrícula 3273JGH para que tenga la matrícula 3233JMG.

UPDATE automoviles SET matricula='3233JMG ' WHERE matricula='3273JGH';

En este caso, dado que la columna matricula de la tabla contratos es **clave ajena o FOREIGN KEY relacionada con las matricula de la tabla contratos con restricción de integridad referencial por actualización en cascada**, además de haberse modificado la matrícula en la tabla automóviles, se habrá modificado la matrícula en todos los contratos correspondientes a ese automóvil.

Ejemplos de uso de UPDATE

Ejemplo 7: Modificar las fechas de los contratos para que en todos aquellos que tengan una fecha inicial superior a la fecha actual, se le reste un año en la fecha que tienen.

```
UPDATE contratos SET fini=subdate(fini,INTERVAL 1 YEAR) WHERE fini>curdate();
```

Ejemplo 8: Modificar las fechas de los contratos para que en todos aquellos en los que la fecha final sea inferior a la inicial, se intercambie el valor de esas fechas.

```
UPDATE contratos SET fini=ffin, ffin=fini WHERE fini>ffin;
```