

INSERT, UPDATE, DELETE

DigitalHouse >
Coding School



**Certified Tech
Developer**
The Ultimate Degree

Índice

1. [INSERT](#)
2. [UPDATE](#)
3. [DELETE](#)

1 | INSERT

INSERT

Existen dos formas de agregar datos en una tabla:

- Insertando datos en **todas** las **columnas**.
- Insertando datos en las **columnas** que **especifiquemos**.

Todas las **columnas**

Si estamos insertando datos en todas las columnas, no hace falta aclarar los nombres de cada columna. Sin embargo, el orden en el que insertemos los valores, deberá ser el mismo orden que tengan asignadas las columnas en la tabla.

```
SQL  INSERT INTO table_name (columna_1, columna_2, columna_3, ...)
      VALUES (valor_1, valor_2, valor_3, ...);
```

```
SQL  INSERT INTO artistas (id, nombre, rating)
      VALUES (DEFAULT, 'Shakira', 1.0);
```

Columnas específicas

Para insertar datos en una columna en específico, aclaramos la tabla y luego escribimos el nombre de la o las columnas entre los paréntesis.

SQL

```
INSERT INTO artistas (nombre)  
VALUES ('Calle 13');
```

SQL

```
INSERT INTO artistas (nombre, rating)  
VALUES ('Maluma', 1.0);
```

2 | UPDATE

UPDATE

UPDATE modificará los registros existentes de una tabla. Al igual que con **DELETE**, es importante no olvidar el **WHERE** cuando escribimos la sentencia, aclarando la condición.

SQL

```
UPDATE nombre_tabla  
SET columna_1 = valor_1, columna_2 = valor_2, ...  
WHERE condición;
```

SQL

```
UPDATE artistas  
SET nombre = 'Charly Garcia', rating = 1.0  
WHERE id = 1;
```


3 | DELETE

DELETE

Con **DELETE** podemos borrar información de una tabla. Es importante recordar utilizar siempre el **WHERE** en la sentencia para agregar la condición de cuáles son las filas que queremos eliminar. Si no escribimos el **WHERE**, estaríamos borrando **toda** la **tabla** y no un registro en particular.

```
SQL  DELETE FROM nombre_tabla WHERE condición;
```

```
SQL  DELETE FROM artistas WHERE id = 4;
```

DigitalHouse>
Coding School