Sql-ALTER TABLE

ALTER TABLE

The `ALTER TABLE` statement is used to modify the structure of an existing table in a database. It is part of the Data Definition Language (DDL) of SQL.

Syntax:

```
 ALTER TABLE table_name
  { ADD | MODIFY | DROP } column_name data_type;
```

or

```
 ALTER TABLE table_name
  RENAME TO new_table_name;

ALTER TABLE table_name
  CHANGE old_column_name new_column_name data_type;

ALTER TABLE table_name
  ADD CONSTRAINT constraint_name constraint_definition;

ALTER TABLE table_name
  DROP CONSTRAINT constraint_name;
```

Examples:

1. Adding a new column

Suppose we have a table called `employees` and we want to add a new column called `salary`.

```
 CREATE TABLE employees (
  id INT PRIMARY KEY,
  name VARCHAR(255),
  department VARCHAR(255)
);

ALTER TABLE employees
ADD salary DECIMAL(10,2);
```

2. Modifying an existing column

Let's say we want to change the data type of the `salary` column from `DECIMAL(10,2)` to `BIGINT`.

```
 ALTER TABLE employees
MODIFY salary BIGINT;
```

3. Dropping a column

Suppose we want to remove the `department` column from the `employees` table.

```
 ALTER TABLE employees
DROP COLUMN department;
```

4. Renaming a table

Let's say we want to rename the `employees` table to `staff`.

```
 ALTER TABLE employees
RENAME TO staff;
```

5. Adding a primary key constraint

Suppose we want to add a primary key constraint called `pk_employees` on the `id` column.

```
 ALTER TABLE employees
ADD CONSTRAINT pk_employees PRIMARY KEY (id);
```

6. Dropping a primary key constraint

Let's say we want to remove the primary key constraint `pk_employees`.

```
 ALTER TABLE employees
DROP CONSTRAINT pk_employees;
```

Note that these examples demonstrate basic usage of the `ALTER TABLE` statement. The actual syntax and behavior may vary depending on the specific database management system (DBMS) being used.

Curated by Brajesh Kumar