

# SQL ALTER TABLE

SQL ALTER TABLE Statement to rename table name, add new column, modify existing column (data type, size, etc.), rename the column, drop the table column.

SQL ALTER TABLE statement is a powerful statement to add, manage or update table structure.

ALTER TABLE Statement to you can do following thing,

- [SQL TABLE RENAME](#)
- [ADD NEW COLUMN IN TABLE](#)
- [MODIFY EXISTING COLUMN IN TABLE](#)
- [RENAME COLUMN IN TABLE](#)
- [DROP THE EXISTING COLUMN IN TABLE](#)

## SQL TABLE RENAME

You can rename the SQL table using this syntax,

### Syntax

```
ALTER TABLE table_name
  RENAME TO new_table_name;
```

### Example

```
SQL> ALTER TABLE userinfo RENAME TO user_info;

Table altered.
```

## SQL ADD NEW COLUMN IN TABLE

You can add new column in table using this syntax,

### Syntax

```
ALTER TABLE table_name
  ADD column_name datatype[(size)];
```

### Example

```
SQL> ALTER TABLE user_info ADD state VARCHAR2(12);

Table altered.
```

## SQL ADD MULTIPLE COLUMN IN TABLE

You can add multiple column in table at a time using this syntax,

### Syntax

```
ALTER TABLE table_name
  ADD ( column_name1 datatype[(size)],
        column_name2 datatype[(size)],
        ...
  );
```

### Example

```
SQL> ALTER TABLE user_info
  ADD (city VARCHAR2(30),
        country VARCHAR2(30)
  );

Table altered.
```

## SQL MODIFY EXISTING COLUMN IN TABLE

You can modify the existing column datatype, size, NOT NULL or CONSTRAINS in table using this syntax,

### Syntax

```
ALTER TABLE table_name
  MODIFY column_name column_datatype[(size)];
```

Example

In this example DESC table structure before execute ALTER statement. ALTER Statement to MODIFY the column size. After alter table again DESC table structure.

```
SQL> DESC user_info;
Name                               Null?   Type
-----
NO                                  NUMBER(3)
NAME                               VARCHAR2(50)
ADDRESS                            VARCHAR2(70)
STATE                              VARCHAR2(12)

SQL> ALTER TABLE user_info MODIFY state VARCHAR2(10);

Table altered.

SQL> DESC user_info;
Name                               Null?   Type
-----
NO                                  NUMBER(3)
NAME                               VARCHAR2(50)
ADDRESS                            VARCHAR2(70)
STATE                              VARCHAR2(12)
```

Same way you can modify multiple existing column in table using following syntax,

Syntax

```
ALTER TABLE table_name
  MODIFY (column_name1 column_datatype[(size)],
         column_name2 column_datatype[(size)],
         ...
);
```

## SQL RENAME COLUMN IN TABLE

You can rename the existing column in table using this syntax,

Syntax

```
ALTER TABLE table_name
  RENAME COLUMN old_column_name TO new_column_name;
```

Example

```
SQL> ALTER TABLE user_info
  RENAME COLUMN no TO sno;

Table altered.
```

## SQL DROP THE COLUMN IN TABLE

You can drop existing column in table using this syntax,

Syntax

```
ALTER TABLE table_name
  DROP COLUMN column_name;
```

Example

This example line 8 country column drop using alter table statement.

SQL> DESC user\_info;

Name	Null?	Type
-----		
SNO		NUMBER(3)
NAME		VARCHAR2(50)
ADDRESS		VARCHAR2(70)
STATE		VARCHAR2(10)
COUNTRY		VARCHAR2(30)

SQL> ALTER TABLE user\_info DROP COLUMN country;

Table altered.

SQL> DESC user\_info;

Name	Null?	Type
-----		
SNO		NUMBER(3)
NAME		VARCHAR2(50)
ADDRESS		VARCHAR2(70)
STATE		VARCHAR2(10)