**Alter Table**

**Add column in table**

Syntax

The syntax to add a column in a table in MySQL (using the ALTER TABLE statement) is:

ALTER TABLE table\_name

ADD new\_column\_name column\_definition

[ FIRST | AFTER column\_name ];

**table\_name**

The name of the table to modify.

**new\_column\_name**

The name of the new column to add to the table.

**column\_definition**

The datatype and definition of the column (NULL or NOT NULL, etc).

**FIRST | AFTER column\_name**

Optional. It tells MySQL where in the table to create the column. If this parameter is not specified, the new column will be added to the end of the table.

Example

Let's look at an example that shows how to add a column in a MySQL table using the ALTER TABLE statement.

For example:

ALTER TABLE contacts

ADD last\_name varchar(40) NOT NULL

AFTER contact\_id;

This MySQL ALTER TABLE example will add a column called *last\_name* to the *contacts* table. It will be created as a NOT NULL column and will appear after the *contact\_id* field in the table.

**Add multiple columns in table**

Syntax

The syntax to add multiple columns in a table in MySQL (using the ALTER TABLE statement) is:

ALTER TABLE table\_name

ADD new\_column\_name column\_definition

[ FIRST | AFTER column\_name ],

ADD new\_column\_name column\_definition

[ FIRST | AFTER column\_name ],

...

;

**table\_name**

The name of the table to modify.

**new\_column\_name**

The name of the new column to add to the table.

**column\_definition**

The datatype and definition of the column (NULL or NOT NULL, etc).

**FIRST | AFTER column\_name**

Optional. It tells MySQL where in the table to create the column. If this parameter is not specified, the new column will be added to the end of the table.

Example

Let's look at an example that shows how to add multiple columns in a MySQL table using the ALTER TABLE statement.

For example:

ALTER TABLE contacts

ADD last\_name varchar(40) NOT NULL

AFTER contact\_id,

ADD first\_name varchar(35) NULL

AFTER last\_name;

This ALTER TABLE example will add two columns to the *contacts* table - *last\_name* and *first\_name*.

The *last\_name* field will be created as a varchar(40) NOT NULL column and will appear after the *contact\_id* column in the table. The *first\_name* column will be created as a varchar(35) NULL column and will appear after the *last\_name* column in the table.

**Modify column in table**

Syntax

The syntax to modify a column in a table in MySQL (using the ALTER TABLE statement) is:

ALTER TABLE table\_name

MODIFY column\_name column\_definition

[ FIRST | AFTER column\_name ];

**table\_name**

The name of the table to modify.

**column\_name**

The name of the column to modify in the table.

**column\_definition**

The modified datatype and definition of the column (NULL or NOT NULL, etc).

**FIRST | AFTER column\_name**

Optional. It tells MySQL where in the table to position the column, if you wish to change its position.

Example

Let's look at an example that shows how to modify a column in a MySQL table using the ALTER TABLE statement.

For example:

ALTER TABLE contacts

MODIFY last\_name varchar(50) NULL;

This ALTER TABLE example will modify the column called *last\_name* to be a data type of varchar(50) and force the column to allow NULL values.

**Modify Multiple columns in table**

Syntax

The syntax to modify multiple columns in a table in MySQL (using the ALTER TABLE statement) is:

ALTER TABLE table\_name

MODIFY column\_name column\_definition

[ FIRST | AFTER column\_name ],

MODIFY column\_name column\_definition

[ FIRST | AFTER column\_name ],

...

;

**table\_name**

The name of the table to modify.

**column\_name**

The name of the column to modify in the table.

**column\_definition**

The modified datatype and definition of the column (NULL or NOT NULL, etc).

**FIRST | AFTER column\_name**

Optional. It tells MySQL where in the table to position the column, if you wish to change its position.

Example

Let's look at an example that shows how to modify multiple columns in a MySQL table using the ALTER TABLE statement.

For example:

ALTER TABLE contacts

MODIFY last\_name varchar(55) NULL

AFTER contact\_type,

MODIFY first\_name varchar(30) NOT NULL;

This ALTER TABLE example will modify two columns to the *contacts* table - *last\_name* and *first\_name*.

The *last\_name* field will be changed to a varchar(55) NULL column and will appear after the *contact\_type* column in the table. The *first\_name* column will be modified to a varchar(30) NOT NULL column (and will not change position in the *contacts* table definition, as there is no *FIRST | AFTER* specified).

**Drop column in table**

Syntax

The syntax to drop a column in a table in MySQL (using the ALTER TABLE statement) is:

ALTER TABLE table\_name

DROP COLUMN column\_name;

**table\_name**

The name of the table to modify.

**column\_name**

The name of the column to delete from the table.

Example

Let's look at an example that shows how to drop a column in a MySQL table using the ALTER TABLE statement.

For example:

ALTER TABLE contacts

DROP COLUMN contact\_type;

This ALTER TABLE example will drop the column called *contact\_type* from the table called *contacts*.

**Rename column in table**

Syntax

The syntax to rename a column in a table in MySQL (using the ALTER TABLE statement) is:

ALTER TABLE table\_name

CHANGE COLUMN old\_name new\_name

column\_definition

[ FIRST | AFTER column\_name ]

**table\_name**

The name of the table to modify.

**old\_name**

The column to rename.

**new\_name**

The new name for the column.

**column\_definition**

The datatype and definition of the column (NULL or NOT NULL, etc). You must specify the column definition when renaming the column, even if it does not change.

**FIRST | AFTER column\_name**

Optional. It tells MySQL where in the table to position the column, if you wish to change its position.

Example

Let's look at an example that shows how to rename a column in a MySQL table using the ALTER TABLE statement.

For example:

ALTER TABLE contacts

CHANGE COLUMN contact\_type ctype

varchar(20) NOT NULL;

This MySQL ALTER TABLE example will rename the column called *contact\_type* to *ctype*. The column will be defined as a varchar(20) NOT NULL column.

**Rename table**

Syntax

The syntax to rename a table in MySQL is:

ALTER TABLE table\_name

RENAME TO new\_table\_name;

**table\_name**

The table to rename.

**new\_table\_name**

The new table name to use.

Example

Let's look at an example that shows how to rename a table in MySQL using the ALTER TABLE statement.

For example:

ALTER TABLE contacts

RENAME TO people;

This ALTER TABLE example will rename the *contacts* table to *people*.