

[Previous](#) [Chapter](#) [Next](#)

## SQL Statements: ALTER TRIGGER to constraint\_clause, 10 of 12

# COMMENT

## Purpose

Use the `COMMENT` statement to add a comment about a table, view, materialized view, or column into the data dictionary.

You can view the comments on a particular table or column by querying the data dictionary views `USER_TAB_COMMENTS`, `DBA_TAB_COMMENTS`, or `ALL_TAB_COMMENTS` or `USER_COL_COMMENTS`, `DBA_COL_COMMENTS`, or `ALL_COL_COMMENTS`.

To drop a comment from the database, set it to the empty string ''.

### See Also:

- [COMMENT](#)
- [Oracle9i Database Reference](#) for information on the data dictionary views

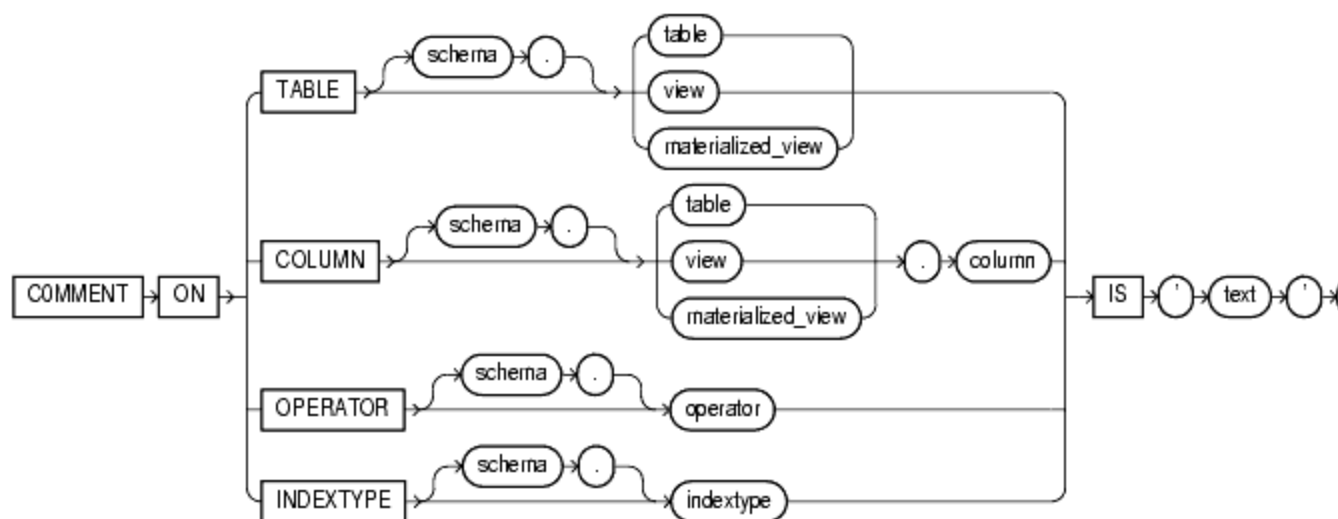
## Prerequisites

The object about which you are adding a comment must be in your own schema or:

- You must have `COMMENT ANY TABLE` system privilege to add a comment to a table, view, or materialized view.
- You must have the `COMMENT ANY INDEXTYPE` system privilege to add a comment to an indextype.
- You must have the `COMMENT ANY OPERATOR` system privilege to add a comment to an operator.

## Syntax

**comment::=**



[Text description of comment](#)

## Keywords and Parameters

### TABLE Clause

Specify the schema and name of the table, view, or materialized view to be commented. If you omit *schema*, Oracle assumes the table, view, or materialized view is in your own schema.

### COLUMN Clause

Specify the name of the column of a table, view, or materialized view to be commented. If you omit *schema*, Oracle assumes the table, view, or materialized view is in your own schema.

### IS 'text'

Specify the text of the comment.

#### See Also:

["Text Literals"](#) for a syntax description of 'text'

### OPERATOR Clause

Specify the name of the operator to be commented. If you omit *schema*, Oracle assumes the operator is in your own schema.

### INDEXTYPE Clause

Specify the name of the indextype to be commented. If you omit *schema*, Oracle assumes the indextype is in your own schema.

## Example

**COMMENT Example**

To insert an explanatory remark on the `job_id` column of the `employees` table, you might issue the following statement:

```
COMMENT ON COLUMN employees.job_id  
  IS 'abbreviated job title';
```

To drop this comment from the database, issue the following statement:

```
COMMENT ON COLUMN employees.job_id IS ' ';
```

---

    
[Previous](#) [Chapter](#) [Next](#)

**ORACLE**  
[Copyright © 1996-2001, Oracle Corporation.](#)  
All Rights Reserved.

       
[Home](#) [Book](#) [Contents](#) [Index](#) [Master Feedback](#)  
[List](#) [Index](#)