The SQL DELETE statement

It is used to delete rows from a table.

Generally DELETE statement removes one or more records from a table.

Syntax

```
DELETE FROM table_name
WHERE condition;
```

> **Note:** Be careful when deleting records in a table! Notice the WHERE clause in the DELETE statement. The WHERE clause specifies which record(s) should be deleted. If you omit the WHERE clause, all records in the table will be deleted!

SQL DELETE Example

```
DELETE FROM Students WHERE Name='Saurav Chaudhary';
```

SQL DELETE TABLE

It is possible to delete all rows in a table without deleting the table. This means that the table structure, attributes, and indexes will remain :-

```
DELETE FROM table_name;
```

Difference between DELETE and TRUNCATE statements

There is a slight difference b/w delete and truncate statement. The **DELETE statement** only deletes the rows from the table based on the condition defined by WHERE clause or delete all the rows from the table when condition is not specified.

But it does not free the space containing by the table.

The **TRUNCATE statement:** it is used to delete all the rows from the table **and free the containing space.**

Difference between DROP and TRUNCATE statements

When you use the drop statement it deletes the table's row together with the table's definition so all the relationships of that table with other tables will no longer be valid.

When you drop a table:

- Table structure will be dropped
- Relationship will be dropped
- Integrity constraints will be dropped
- Access privileges will also be dropped

On the other hand when we **TRUNCATE** a table, the table structure remains the same, so you will not face any of the above problems.

SQL DELETE VIEW

Before deleting a view we should know about what a view is.

In SQL, a view is a virtual table based on the result-set of an SQL statement.

A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.

You can add SQL functions, WHERE, and JOIN statements to a view and present the data as if the data were coming from one single table.

Create View

Syntax

```
CREATE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition
```

> **Note:** A view always shows up-to-date data! The database engine recreates the data, using the view's SQL statement, every time a user queries a view.

SQL CREATE VIEW Examples

```
CREATE VIEW Student AS
SELECT Reg_NO, Branch
FROM Students
WHERE Blood_Group='B+';

Select * From Student;
```

Update View

Syntax

```
CREATE OR REPLACE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

To delete we use Drop view

Syntax

```
DROP VIEW view_name;
```

Example

```
DROP VIEW Student;
```