



# **DELETE Statement in detail**



# DELETE statement

- The MySQL DELETE statement is used to delete the existing records in a table.
- The DELETE statement is one of the primary data modification language (DML) statements available in Transact-SQL, along with INSERT, MERGE, and UPDATE.
- WHERE clause with a DELETE query to delete the selected rows, otherwise all the records would be deleted.
- Of all the DML statements, the DELETE statement is probably the easiest to use and most POWERFUL.
- DELETE statement can be used to delete one row, multiple rows or delete entire table.



# Basic DELETE Statement

- DELETE FROM tablename;
- This statement will delete all the rows in this table.
- DELETE statement supports the optional FROM keyword, which is inserted between the DELETE keyword and the name of the table
- Remember DELETE works on the rows not on the columns.



# DELETE statement with where clause

- To limit the rows that are updated when you issue an DELETE statement, add a WHERE clause
- The WHERE clause specifies the search conditions that define which rows in the target table should be deleted.
- `DELETE FROM SalesStaff WHERE CountryRegion = 'United States';`
- Another way you can remove only a subset of rows from a table is to include the LIMIT clause.
- `DELETE FROM SalesStaff LIMIT 20;`



# DELETE using data from other table

- There will be times when you'll want to delete rows from a table based on data in another table.
- One approach is to create a subquery that retrieves data from the other table.
- `DELETE SalesStaff WHERE StaffID IN (SELECT BusinessEntityID FROM Sales.vSalesPerson WHERE SalesLastYear = 0);`
- Another method you can use to achieve the same results is to join between the target table and the table that contains the lookup information.
- `DELETE SalesStaff FROM Sales.vSalesPerson sp INNER JOIN dbo.SalesStaff ss ON sp.BusinessEntityID = ss.StaffID WHERE sp.SalesLastYear = 0;`