

SQL Statement	Syntax
AND / OR	<pre>SELECT column_name(s) FROM table_name WHERE condition AND OR condition</pre>
ALTER TABLE	<pre>ALTER TABLE table_name ADD column_name datatype or ALTER TABLE table_name DROP COLUMN column_name</pre>
AS (alias)	<pre>SELECT column_name AS column_alias FROM table_name or SELECT column_name FROM table_name AS table_alias</pre>
BETWEEN	<pre>SELECT column_name(s) FROM table_name WHERE column_name BETWEEN value1 AND value2</pre>
CREATE DATABASE	<pre>CREATE DATABASE database_name</pre>
CREATE TABLE	<pre>CREATE TABLE table_name (column_name1 data_type, column_name2 data_type, column_name3 data_type, ...)</pre>
CREATE INDEX	<pre>CREATE INDEX index_name ON table_name (column_name) or CREATE UNIQUE INDEX index_name ON table_name (column_name)</pre>
CREATE VIEW	<pre>CREATE VIEW view_name AS SELECT column_name(s)</pre>

	FROM table_name WHERE condition
DELETE	DELETE FROM table_name WHERE some_column=some_value or DELETE FROM table_name (Note: Deletes the entire table!!) DELETE * FROM table_name (Note: Deletes the entire table!!)
DROP DATABASE	DROP DATABASE database_name
DROP INDEX	DROP INDEX table_name.index_name (SQL Server) DROP INDEX index_name ON table_name (MS Access) DROP INDEX index_name (DB2/Oracle) ALTER TABLE table_name DROP INDEX index_name (MySQL)
DROP TABLE	DROP TABLE table_name
EXISTS	IF EXISTS (SELECT * FROM table_name WHERE id = ?) BEGIN --do what needs to be done if exists END ELSE BEGIN --do what needs to be done if not END
GROUP BY	SELECT column_name, aggregate_function(column_name) FROM table_name WHERE column_name operator value GROUP BY column_name
HAVING	SELECT column_name, aggregate_function(column_name) FROM table_name WHERE column_name operator value GROUP BY column_name HAVING aggregate_function(column_name) operator value
IN	SELECT column_name(s) FROM table_name

	WHERE column_name IN (value1,value2,..)
INSERT INTO	INSERT INTO table_name VALUES (value1, value2, value3,...) <i>or</i> INSERT INTO table_name (column1, column2, column3,...) VALUES (value1, value2, value3,...)
INNER JOIN	SELECT column_name(s) FROM table_name1 INNER JOIN table_name2 ON table_name1.column_name=table_name2.column_name
LEFT JOIN	SELECT column_name(s) FROM table_name1 LEFT JOIN table_name2 ON table_name1.column_name=table_name2.column_name
RIGHT JOIN	SELECT column_name(s) FROM table_name1 RIGHT JOIN table_name2 ON table_name1.column_name=table_name2.column_name
FULL JOIN	SELECT column_name(s) FROM table_name1 FULL JOIN table_name2 ON table_name1.column_name=table_name2.column_name
LIKE	SELECT column_name(s) FROM table_name WHERE column_name LIKE pattern
ORDER BY	SELECT column_name(s) FROM table_name ORDER BY column_name [ASC DESC]
SELECT	SELECT column_name(s) FROM table_name
SELECT *	SELECT * FROM table_name
SELECT DISTINCT	SELECT DISTINCT column_name(s) FROM table_name

SELECT INTO	<pre>SELECT * INTO new_table_name [IN externaldatabase] FROM old_table_name or SELECT column_name(s) INTO new_table_name [IN externaldatabase] FROM old_table_name</pre>
SELECT TOP	<pre>SELECT TOP number percent column_name(s) FROM table_name</pre>
TRUNCATE TABLE	<pre>TRUNCATE TABLE table_name</pre>
UNION	<pre>SELECT column_name(s) FROM table_name1 UNION SELECT column_name(s) FROM table_name2</pre>
UNION ALL	<pre>SELECT column_name(s) FROM table_name1 UNION ALL SELECT column_name(s) FROM table_name2</pre>
UPDATE	<pre>UPDATE table_name SET column1=value, column2=value,... WHERE some_column=some_value</pre>
WHERE	<pre>SELECT column_name(s) FROM table_name WHERE column_name operator value</pre>

Source : https://www.w3schools.com/sql/sql_quickref.asp