



LEARN SQLITE

sql database engine

SQLite Tutorial

- [SQLite - Home](#)
- [SQLite - Overview](#)
- [SQLite - Installation](#)
- [SQLite - Commands](#)
- [SQLite - Syntax](#)
- [SQLite - Data Type](#)
- [SQLite - CREATE Database](#)
- [SQLite - ATTACH Database](#)
- [SQLite - DETACH Database](#)
- [SQLite - CREATE Table](#)
- [SQLite - DROP Table](#)
- [SQLite - INSERT Query](#)
- [SQLite - SELECT Query](#)
- [SQLite - Operators](#)
- [SQLite - Expressions](#)
- [SQLite - WHERE Clause](#)
- [SQLite - AND & OR Clauses](#)
- [SQLite - UPDATE Query](#)
- [SQLite - DELETE Query](#)
- [SQLite - LIKE Clause](#)
- [SQLite - GLOB Clause](#)
- [SQLite - LIMIT Clause](#)
- [SQLite - ORDER By Clause](#)
- [SQLite - GROUP By Clause](#)
- [SQLite - HAVING Clause](#)
- [SQLite - DISTINCT Keyword](#)

Advanced SQLite

- [SQLite - PRAGMA](#)
- [SQLite - Constraints](#)
- [SQLite - JOINS](#)
- [SQLite - UNIONS Clause](#)
- [SQLite - NULL Values](#)
- [SQLite - ALIAS Syntax](#)
- [SQLite - Triggers](#)
- [SQLite - Indexes](#)
- [SQLite - INDEXED By Clause](#)
- [SQLite - ALTER Command](#)
- [SQLite - TRUNCATE Command](#)
- [SQLite - Views](#)
- [SQLite - Transactions](#)
- [SQLite - Subqueries](#)
- [SQLite - AUTOINCREMENT](#)
- [SQLite - Injection](#)
- [SQLite - EXPLAIN](#)
- [SQLite - VACUUM](#)
- [SQLite - Date & Time](#)
- [SQLite - Useful Functions](#)

SQLite Interfaces

- [SQLite - C/C++](#)
- [SQLite - Java](#)
- [SQLite - PHP](#)
- [SQLite - Perl](#)
- [SQLite - Python](#)

SQLite Useful Resources

- [SQLite - Quick Guide](#)
- [SQLite - Useful Resources](#)
- [SQLite - Discussion](#)

Selected Reading

- [UPSC IAS Exams Notes](#)
- [Developer's Best Practices](#)
- [Questions and Answers](#)
- [Effective Resume Writing](#)
- [HR Interview Questions](#)
- [Computer Glossary](#)
- [Who is Who](#)

SQLite - Expressions

Advertisements



⏪ Previous Page

Next Page ⏩

An expression is a combination of one or more values, operators, and SQL functions that evaluate to a value.

SQL expressions are like formulas and they are written in query language. You can also use to query the database for a specific set of data.

Syntax

Consider the basic syntax of the SELECT statement as follows –

```
SELECT column1, column2, columnN
FROM table_name
WHERE [CONDITION | EXPRESSION];
```

Following are the different types of SQLite expressions.

SQLite - Boolean Expressions

SQLite Boolean Expressions fetch the data on the basis of matching single value. Following is the syntax –

```
SELECT column1, column2, columnN
FROM table_name
WHERE SINGLE VALUE MATCHING EXPRESSION;
```

Consider COMPANY table with the following records –

ID	NAME	AGE	ADDRESS	SALARY
1	Paul	32	California	20000.0
2	Allen	25	Texas	15000.0
3	Teddy	23	Norway	20000.0
4	Mark	25	Rich-Mond	65000.0
5	David	27	Texas	85000.0
6	Kim	22	South-Hall	45000.0
7	James	24	Houston	10000.0

Following is a simple examples showing the usage of SQLite Boolean Expressions –

```
sqlite> SELECT * FROM COMPANY WHERE SALARY = 10000;
```

ID	NAME	AGE	ADDRESS	SALARY
4	James	24	Houston	10000.0

SQLite - Numeric Expression

These expressions are used to perform any mathematical operation in any query. Following is the syntax –

```
SELECT numerical_expression as OPERATION_NAME
[FROM table_name WHERE CONDITION] ;
```

Here, numerical\_expression is used for mathematical expression or any formula. Following is a simple example showing the usage of SQLite Numeric Expressions.

```
sqlite> SELECT (15 + 6) AS ADDITION
ADDITION = 21
```

There are several built-in functions such as **avg()**, **sum()**, **count()**, etc., to perform what is known as **aggregate data calculations** against a table or a specific table column.

```
sqlite> SELECT COUNT(*) AS "RECORDS" FROM COMPANY;
RECORDS = 7
```

SQLite - Date Expressions

Date Expressions returns the current system date and time values. These expressions are used in various data manipulations.

```
sqlite> SELECT CURRENT_TIMESTAMP;
CURRENT_TIMESTAMP = 2013-03-17 10:43:35
```

⏪ Previous Page

🖨 Print Page

Next Page ⏩

Advertisements

