



SQL | Date Functions (Set-2)



Sakshi98

[Read](#)[Discuss](#)[Courses](#)[Practice](#)

In SQL, dates are complicated for newbies, since while working with a database, the format of the date in the table must be matched with the input date in order to insert. In various scenarios instead of date, datetime (time is also involved with date) is used.

Some of the date functions have been already discussed in the [Set-1](#). In this post, the remaining date functions have been discussed.

Below are the remaining date functions that are used in SQL:

1. **MICROSECOND():** It returns the microsecond portion of a date value.

Syntax: `SELECT MICROSECOND("2018-07-18 09:12:00.000345");`

Output: 345

2. **MINUTE():** It returns the minute portion of a date value.

Syntax: `SELECT MINUTE("2018-07-18 09:12:00");`

Output: 12

3. **MONTH():** It returns the month portion of a date value.

Syntax: `SELECT MONTH ('2018/07/18')AS MONTH;`

Output: 7

4. **MONTHNAME():** It returns the full month name for a date.

Syntax: `SELECT MONTHNAME("2018/07/18");`

Output: JULY

5. **NOW():** It returns the current date and time.

Syntax: `SELECT NOW();`

Output: 2018-07-18 09:14:32

6. **PERIOD_ADD():** It takes a period and adds a specified number of months to it.

Syntax: `SELECT PERIOD_ADD(201803, 6);`

Output: 201809

7. **PERIOD_DIFF()**: It returns the difference in months between two periods.

Syntax: `SELECT PERIOD_DIFF(201810, 201802);`

Output: 8

8. **QUARTER()**: It returns the quarter portion of a date value.

Syntax: `SELECT QUARTER("2018/07/18");`

Output: 3

9. **SECOND()**: It returns the second portion of a date value.

Syntax: `SELECT SECOND("09:14:00:00032");`

Output: 0

10. **SEC_TO_TIME()**: It converts numeric seconds into a time value.

Syntax: `SELECT SEC_TO_TIME(1);`

Output: 00:00:01

11. **STR_TO_DATE()**: It takes a string and returns a date specified by a format mask.

Syntax: `SELECT STR_TO_DATE("JULY 18 2018", "%M %D %Y");`

Output: 0018-07-18

12. **SUBDATE()**: It returns a date after which a certain time/date interval has been subtracted.

Syntax: `SELECT SUBDATE("2017-06-15", INTERVAL 10 DAY);`

Output: 2017-06-05

13. **SUBTIME()**: It returns a time/date time value after a certain time interval has been subtracted.

Syntax: `SELECT SUBDATE("2018/07/18", INTERVAL 10 DAY);`

Output: 2018-07-18 09:15:17.542768

14. **SYSDATE()**: It returns the current date and time.

Syntax: `SELECT SYSDATE();`

Output: 2018-07-18 09:19:03

15. **TIME()**: It extracts the time value from a time/date time expression.

Syntax: `SELECT TIME("09:16:10");`

Output: 09:16:10

16. **TIME_FORMAT():** It formats the time as specified by a format mask.

Syntax: `SELECT TIME_FORMAT("09:16:10", "%H %I %S");`

Output: 09 09 10

17. **TIME_TO_SEC():** It converts a time value into numeric seconds.

Syntax: `SELECT TIME_TO_SEC("09:16:10");`

Output: 33370

18. **TIMEDIFF():** It returns the difference between two time/datetime values.

Syntax: `SELECT TIMEDIFF("09:16:10", "09:16:04");`

Output: 00:00:06

19. **TIMESTAMP():** It converts an expression to a date time value and if specified adds an optional time interval to the value.

Syntax: `SELECT TIMESTAMP("2018-07-18", "09:16:10");`

Output: 2018-07-18 09:16:10

20. **TO_DAYS():** It converts a date into numeric days.

Syntax: `SELECT TO_DAYS("2018-07-18");`

Output: 737258

21. **WEEK():** It returns the week portion of a date value.

Syntax: `SELECT WEEK("2018-07-18");`

Output: 28

22. **WEEKDAY():** It returns the weekday index for a date value.

Syntax: `SELECT WEEKDAY("2018-07-18");`

Output: 2

23. **WEEKOFYEAR():** It returns the week of the year for a date value.

Syntax: `SELECT WEEKOFYEAR("2018-07-18");`

Output: 29

24. **YEAR():** It returns the year portion of a date value.

Syntax: `SELECT YEAR("2018-07-18");`

Output: 2018

25. **YEARWEEK():** It returns the year and week for a date value.

Syntax: `SELECT YEARWEEK("2018-07-18");`

Output: 201828

Last Updated : 19 Jul, 2018

4

Similar Reads

1. [SQL | Date Functions \(Set-1\)](#)
2. [How to Write a SQL Query For a Specific Date Range and Date Time?](#)
3. [SQL | Date functions](#)
4. [Useful Date and Time Functions in PL/SQL](#)
5. [Date manipulating functions in SQL](#)
6. [SQL Query to Check if Date is Greater Than Today in SQL](#)
7. [SQL | Functions \(Aggregate and Scalar Functions\)](#)
8. [SQL | Difference between functions and stored procedures in PL/SQL](#)
9. [Difference between Structured Query Language \(SQL\) and Transact-SQL \(T-SQL\)](#)
10. [Configure SQL Jobs in SQL Server using T-SQL](#)

[Previous](#)

[Next](#)

Article Contributed By :



Sakshi98
Sakshi98

Vote for difficulty

Current difficulty : [Easy](#)

Easy

Normal

Medium

Hard

Expert

Article Tags : [SQL-Functions](#), [SQL](#)

Practice Tags : [SQL](#)