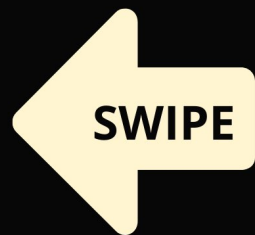




#ASLI ENGINEERING

MySQL DATE, DATETIME, and TIMESTAMP



BY

ARPIT BHAYANI

Datetime and Timestamp

Which one to use and when?

← Restrictive

DATE : stores date but not time (1000-01-01 to 9999-12-31)

DATETIME : stores date and time (1000-01-01 00:00:00 to 9999-12-31 23:59:59)

TIMESTAMP : stores date and time (unix epoch as integer)
(1970-01-01 00:00:00 UTC to 2038-01-19 03:14:07 UTC)

↳ 0 to 2^{32} (seconds)

What if we want to store at microseconds level granularity?

↳ DATETIME and TIMESTAMP supports microseconds (additional)

↓
1000-01-01 00:00:00.000000 to
9999-12-31 23:59:59.999999

Fractional seconds take up
0 to 3 bytes depending on precision

Storage Requirements

0 precision → 0 bytes
1, 2 precision → 1 byte

DATE	3 bytes
DATETIME	5 bytes + fractional seconds
TIMESTAMP	4 bytes + fractional seconds

DATETIME

convenience

Use DATETIME when you want to store usecase specific time

eg: appointment, schedule_at (specific and static)

When you want to do calculations within MySQL

eg: DATE_ADD(purchased_at, 'INTERVAL 1 DAY')

DATETIME is stored as a compact 5 byte representation (MySQL)

but covering a longer range than timestamp

Human-readable and native language object support

↳ Queries and Console output

TIMESTAMP

Timestamps are stored as integers representing time in UTC

* timezone conversion happens during return & accepting

* on disk it is just UTC

return value changes with the connection timezone

Timestamps are light weight than Datetime

↳ on storage and index (1 byte better)

Use timestamps when you want to record system time

eg: created_at, transaction time, etc