bkie

Data Types in MySQL

BINARY

- gobs of data - 1s & 0s
- TL;DR: every two hexits in the UUID can go to 1 byte in binary
- SAVE BIG! 50% savings compared to a string type
- so we will use BINARY(16) to store UUIDs for PKs/FKs
- also, binary data is faster to search than strings

https://dev.mysql.com/doc/refman/8.0/en/binary-varbinary.html

CHAR & VARCHAR

- CHAR(8) vs VARCHAR(8) - Example: fuzzy
- CHAR : F U Z Z Y / / / 3 bytes wasted
- VARCHAR : F U Z Z Y (nul) // no bytes wasted, nul is a stop byte, k + 1
- CHAR is faster, VARCHAR is more flexible
- CHAR <= 255 and VARCHAR <= 65535
- also, expanded VARCHARs don't index well

https://dev.mysql.com/doc/refman/8.0/en/char.html

Int

- SIGNED vs UNSIGNED : allow/disallow negatives
- mySQL has 5 different sizes:
  - TINYINT
    - SIGNED [-128, 127]
    - UNSIGNED [0, 255]
  - SMALLINT
  - MEDIUMINT
  - INT
  - BIGINT

https://dev.mysql.com/doc/refman/8.0/en/integer-types.html

Float

- DECIMAL(s, d) will give you a number with s significant figures and d decimal places, where s >= d
- e.g. DECIMAL(5, 3): . _ [-99.999, 99.999]
- DECIMAL(7, 2: . _ [-99999.99, 99999.99]

https://dev.mysql.com/doc/refman/8.0/en/floating-point-types.html

Date

- beginning of time: 1970-01-01 00:00:00 UTC
- UTC is Europe/London without Daylight Savings
- DATETIME(6) : Y-m-d H:i:s.u with 6 decimal places
- DATE : same without time
- TIME : same without date
- YEAR : WTF? SQL can only store >= 1980, use an integer
- TIMESTAMP(6) : same as DATETIME(6) except mySQL automagically fills it in for you on INSERT and/or UPDATE - INSERT only by default

https://dev.mysql.com/doc/refman/8.0/en/date-and-time-types.html