Juggling Data Types

---

1. Learn: Understanding SQL Data Types

SQL data types define the kind of data each column can store. Choosing appropriate types:

- Saves storage space
- Ensures correct data formats
- Improves performance

A. Numeric Data Types

Exact Numeric Types

- INT or INTEGER: Whole numbers. Used for things like IDs, counters, quantities.
- SMALLINT: A smaller range of integers. Useful when values are known to be small (e.g., status codes).
- BIGINT: Larger range than INT, ideal for high-scale identifiers.
- DECIMAL(p, s) or NUMERIC(p, s): Fixed precision decimal values. Use for currency and precise calculations.
    - p: Total number of digits
    - s: Digits after the decimal point

Approximate Numeric Types

- FLOAT: Floating-point number with approximate precision. Good for scientific data where minor inaccuracies are okay.
- REAL: Similar to FLOAT, with less precision.
- DOUBLE PRECISION: Higher precision than FLOAT, used in complex calculations.

B. String (Character) Data Types

- CHAR(n): Fixed-length string. Pads the value with spaces if shorter than n. Ideal for fixed formats like codes or flags.
- VARCHAR(n): Variable-length string with a maximum of n characters. Best for names, emails, and addresses.
- TEXT: Can store long text entries without a fixed limit. Useful for descriptions, articles, or logs. Not ideal for indexing.

C. Date and Time Data Types

- DATE: Stores only the date in format YYYY-MM-DD.
- TIME: Stores only time in format HH:MM:SS.
- DATETIME: Stores both date and time.
- TIMESTAMP: Similar to DATETIME but often auto-generates the current time. Useful for logging.
- YEAR: Stores just the year. Useful for year-based data.

D. Boolean Type

- BOOLEAN or BOOL: Stores TRUE or FALSE values. Often stored as 1 (true) and 0 (false) in databases.

E. Binary Data Types

Used for non-text content like files, images, or encrypted data.

BINARY(n): Fixed-length binary storage.

VARBINARY(n): Variable-length binary data.

BLOB: Binary Large Object. Used to store multimedia like images or PDFs.

2. Practice: Try It Yourself

Create a Table with Mixed Data Types

```
CREATE TABLE Products (
    product_id INT PRIMARY KEY,
    product_name VARCHAR(100),
    price DECIMAL(10, 2),
    description TEXT,
    available BOOLEAN,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Insert Sample Records

```
INSERT INTO Products
(product_id, product_name, price, description, available)
VALUES
(1, 'Laptop', 599.99, 'Lightweight and fast', TRUE),
(2, 'Mouse', 19.99, 'Wireless mouse', TRUE);
```

View Table Structure

```
DESCRIBE Products;
SELECT column_name, data_type
FROM information_schema.columns
WHERE table_name = 'Products';
```

8. FAQ

Q1: What's the difference between VARCHAR and TEXT?
A: VARCHAR has a defined maximum length and supports indexing. TEXT is used for large blocks of text but may not support indexing efficiently.

Q2: When should I use DECIMAL instead of FLOAT?
A: Use DECIMAL when exact precision is needed (e.g., currency). FLOAT is suitable for approximate values like scientific measurements.

Q3: Can I store both date and time in one column?
A: Yes, use DATETIME or TIMESTAMP. TIMESTAMP is often preferred for automatic current time tracking.

Q4: What is the max length for VARCHAR?
A: It depends on the DBMS. MySQL can allow up to 65,535 bytes in a row, but practical limits for VARCHAR are often around 255–1000 characters.

Q5: Why not use TEXT for all text fields?
A: TEXT fields can't be efficiently indexed or sorted in many databases. Use VARCHAR if you need to filter or search the text.