SQL Data Types and Constraints

What is a Data Type?

A data type defines the kind of data that can be stored in a table column. Data types help with data validation, storage optimization, and query performance in relational databases. They help keep data accurate, efficient, and easy to manipulate within SQL databases.

Types of Data Types

Numeric:

INT: Integer values (e.g., 1, 100).
FLOAT/REAL: Floating-point numbers (e.g., 3.14).
DECIMAL/NUMERIC: Fixed precision numbers (e.g., 123.45).

Character/String:

CHAR(n): Fixed-length string (e.g., 'ABC').
VARCHAR(n): Variable-length string (e.g., 'Hello').
TEXT: Large text data.

Date and Time:

DATE: Stores date (e.g., '2025-12-01').
TIME: Stores time (e.g., '14:30:00').
DATETIME: Stores both date and time.

Other Data Types:

BOOLEAN: True/False values.
BLOB: Binary large objects (e.g., images, files).
ENUM: Allows only one value from predefined list.
SET: Stores multiple values from predefined list.

Constraints in SQL

Constraints are rules applied to columns or tables to enforce data integrity, accuracy, and reliability by restricting the type of data that can be inserted or manipulated.

Types of SQL Constraints:

NOT NULL

- Ensures a column cannot have NULL (empty) values.

```
CREATE TABLE students (
 id INT,
 name VARCHAR(100) NOT NULL
);
```

UNIQUE

- Ensures all values in a column are distinct with no duplicates.

```
CREATE TABLE employees (
 emp_id INT PRIMARY KEY,
 email VARCHAR(150) UNIQUE
);
```

PRIMARY KEY

- Uniquely identifies each row; combines NOT NULL and UNIQUE.

```
CREATE TABLE products (
 product_id INT PRIMARY KEY,
 name VARCHAR(100),
 price DECIMAL(10,2)
);
```

FOREIGN KEY

- It creates a relationship between two tables and helps maintain referential integrity.

```
CREATE TABLE orders (
order_id INT PRIMARY KEY,
product_id INT,
FOREIGN KEY (product_id) REFERENCES products(product_id)
);
```

CHECK

- Validates that data meets a specific condition.

```
CREATE TABLE orders (
order_id INT PRIMARY KEY,
amount INT CHECK (amount >= 100)
);
```

DEFAULT

- It sets a default value if value was is provided.

```
CREATE TABLE employees (
id INT PRIMARY KEY,
salary DECIMAL(10,2) DEFAULT 50000.00
);
```

INDEX

- Speeds up data searching (NOT for data integrity).

```
CREATE INDEX idx_name ON Employee(EmpName);
```