MySQL Data Types

MySQL uses many different data types broken into three categories −

- Numeric
- Date and Time
- String Types.

Below given are some important data types from all the 3 categories:

- **INT** − A normal-sized integer that can be signed or unsigned. If signed, the allowable range is from -2147483648 to 2147483647. If unsigned, the allowable range is from 0 to 4294967295. You can specify a width of up to 11 digits.
- **BIGINT** − A large integer that can be signed or unsigned. If signed, the allowable range is from -9223372036854775808 to 9223372036854775807. If unsigned, the allowable range is from 0 to 18446744073709551615. You can specify a width of up to 20 digits.

- **FLOAT(M,D)** − A floating-point number that cannot be unsigned. You can define the display length (M) and the number of decimals (D). This is not required and will default to 10,2, where 2 is the number of decimals and 10 is the total number of digits (including decimals). Decimal precision can go to 24 places for a FLOAT.
- **DECIMAL(M,D)** − An unpacked floating-point number that cannot be unsigned. In the unpacked decimals, each decimal corresponds to one byte. Defining the display length (M) and the number of decimals (D) is required. NUMERIC is a synonym for DECIMAL
- **DATE** − A date in YYYY-MM-DD format, between 1000-01-01 and 9999-12-31. For example, December 30th, 1973 would be stored as 1973-12-30.
- **CHAR(M)** − A fixed-length string between 1 and 255 characters in length (for example CHAR(5)), right-padded with spaces to the specified length when stored. Defining a length is not required, but the default is 1.
- **VARCHAR(M)** − A variable-length string between 1 and 255 characters in length. For example, VARCHAR(25). You must define a length when creating a VARCHAR field.

MySQL DATA TYPES

DATE TYPE	SPEC	DATA TYPE	SPEC
CHAR	String (0 - 255)	INT	Integer (-2147483648 to 214748-3647)
VARCHAR	String (0 - 255)	BIGINT	Integer (-9223372036854775808 to 9223372036854775807)
TINYTEXT	String (0 - 255)	FLOAT	Decimal (precise to 23 digits)
TEXT	String (0 - 65535)	DOUBLE	Decimal (24 to 53 digits)
BLOB	String (0 - 65535)	DECIMAL	"DOUBLE" stored as string
MEDIUMTEXT	String (0 - 16777215)	DATE	YYYY-MM-DD
MEDIUMBLOB	String (0 - 16777215)	DATETIME	YYYY-MM-DD HH:MM:SS
LONGTEXT	String (0 - 4294967295)	TIMESTAMP	YYYYMMDDHHMMSS
LONGBLOB	String (0 - 4294967295)	TIME	HH:MM:SS
TINYINT	Integer (-128 to 127)	ENUM	One of preset options
SMALLINT	Integer (-32768 to 32767)	SET	Selection of preset options
MEDIUMINT	Integer (-8388608 to 8388607)	BOOLEAN	TINYINT(1)

CHAR v/s VARCHAR

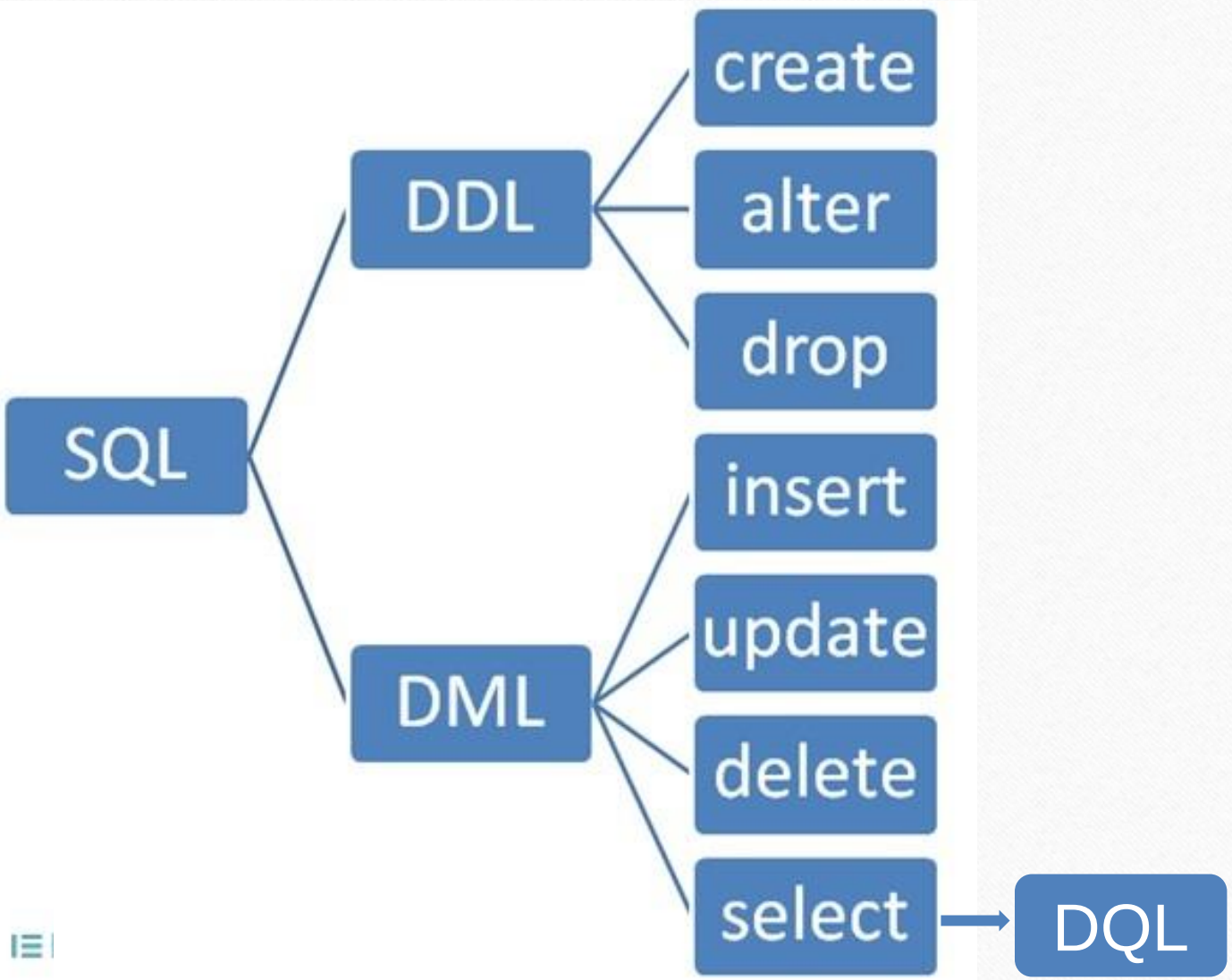
- CHAR is used to store character string of fixed length specified. If the length of string is less than set or fixed length then it is padded with extra blank spaces so that its length became equal to the set length. We should use this datatype when we expect the data values in a column are of same length.
- VARCHAR is used to store character string of variable. If the length of string is less than specified length then it will store as it is without padded with extra blank spaces. Storage size of VARCHAR datatype is equal to the actual length of the entered string in bytes.
- Consider CHAR(10) and VARCHAR(10), if we enter **VEDANT** then CHAR will still use 10 character space where VARCHAR will use only 6 and rest will be freed.

Types of SQL Commands

- **DDL** (Data Definition Language) To create database and table structure- commands like CREATE , ALTER , DROP etc.
- **DML** (Data Manipulation Language) Record/rows related operations. commands like INSERT..., DELETE..., UPDATE.... etc.
- **DQL** (Data Query Language) To retrieve the data from database SELECT
- **DCL** (Data Control Language) used to manipulate permissions or access rights to the tables. commands like GRANT , REVOKE etc.
- **TCL**

Database Handling Commands

- **Creating a Database.**
  - The following command will create School database in MySQL.
  - mysql> CREATE DATABASE School;
- **Opening a database**
  - To open an existing database, following command is used.
  - mysql> USE school;
- **Listing of database and tables**
  - mysql> SHOW DATABASES;
  - mysql> SHOW TABLES;
- **Viewing Table Structure**
  - mysql> DESCRIBE Student;


SQL
DDL
create
alter
drop
DML
insert
update
delete
select
DQL

DDL & DML

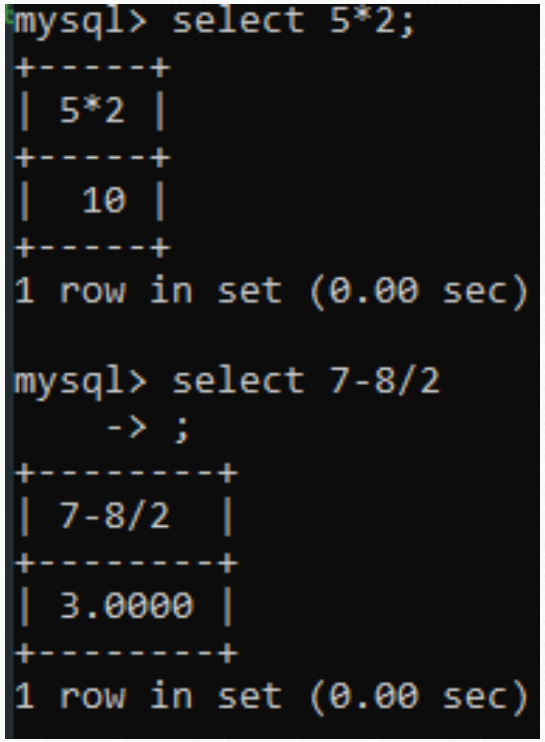
- **CREATE** – is used to create the database or its objects (like table, index, function, views, store procedure and triggers).
- **DROP** – is used to delete objects from the database.
- **ALTER**- is used to alter the structure of the database.

- **SELECT** – is used to retrieve data from the a database.
- **INSERT** – is used to insert data into a table.
- **UPDATE** – is used to update existing data within a table.
- **DELETE** – is used to delete records from a database table.

Expressions and Operators

- Arithmetic Operators : **+-*/**
- Comparison Operators : **=,<,>,<=,>=,<>**
- Logical Operators : **AND, OR, NOT**

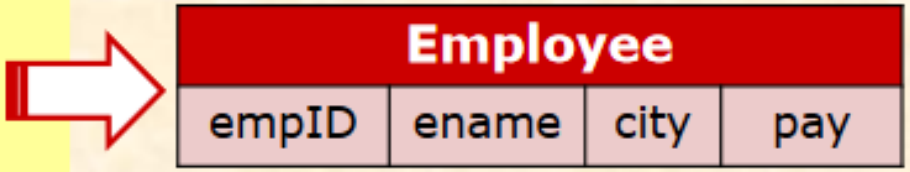
- **Expressions:** SQL expression is a combination of one or more values, operators



Create Table (DDL)

- **Creating Simple Tables:**
  - **CREATE TABLE < Table Name>(<Col name1><data type> [(size)] [constraints],....);**
  - Data types-INTEGER, NUMERIC(P,D), CHAR(n), VARCHAR(n), DATE etc.

```
mysql> CREATE TABLE Employee
       (empID integer,
        ename char(30),
        city char(25),
        pay decimal(10,2));
```



ALTER TABLE (DDL)

- In SQL if we ever need to change the structure of the database then ALTER TABLE command is used. By using this command we can add a column in the existing table, delete a column from a table or modify columns in a table.

ALTER TABLE <TABLENAME>

ADD/DROP/MODIFY/RENAME <details>

ALTER TABLE (DDL)

- **ALTER TABLE <Table Name>**
  ADD/MODIFY/RENAME/DROP<Column Details>
- **ADD -** Add a new Column or Constraints
- **MODIFY -** Modifying existing column (name, data type, size etc.)
- **DROP -** Delete an existing column or Constraints
- **RENAME -** Changing Column Name

ALTER TABLE- Adding a column

- The syntax to add a column is:-
- ALTER TABLE <table_name> ADD <column_name datatype>;

- e.g **ALTER TABLE student ADD Class INT;**
- The above command add a column Address to the table student.

ALTER TABLE- Removing a column

- The syntax to add a column is:-
- ALTER TABLE <table_name> DROP [COLUMN] <column_name>;

- e.g. **ALTER TABLE Student DROP COLUMN City;**

  Or ALTER TABLE Student DROP City;
- The column City will be removed from the table student

ALTER TABLE- MODIFY

- MODIFY is used along with ALTER TABLE to change existing column definition(data type, constraint etc.)
- ALTER TABLE <*table_name*> MODIFY [COLUMN] <*column_name*> <*new datatype*>;

E.g.: **ALTER TABLE STUDENT MODIFY COLUMN CLASS VARCHAR(10)**

Or ALTER TABLE STUDENT MODIFY CLASS VARCHAR(10)

- **CLASS** column is modified with new datatype VARCHAR(10)

ALTER TABLE - RENAME

- **We can rename table using alter**
- **ALTER TABLE <Old Table Name> RENAME <New Table Name>**
- **Examples:**
  - **ALTER TABLE STUDENT RENAME NEWSTUDENT;**
  - **ALTER TABLE EMP RENAME EMPLOYEE;**
- **Alternative Method:**
- **RENAME TABLE <old_table_name> TO <new_table_name>;**

ALTER TABLE- RENAME

- RENAME COLUMN is used along with ALTER TABLE to change existing column definition
- ALTER TABLE <table_name> RENAME COLUMN <old_name> TO <new_name>;

  E.g.: **ALTER TABLE STUDENT RENAME COLUMN RollNo TO Roll_Num;**

- **MySQL support CHANGE keyword also**
- ALTER TABLE <table_name> CHANGE<old_name> <new_name><data_type>;

DROP TABLE (DDL)

- Sometimes you may need to drop a table which is not in use. DROP TABLE command is used to Delete / drop a table permanently. It should be kept in mind that we can not drop a table if it contains records. That is first all the rows of the table have to be deleted and only then the table can be dropped.

- The general syntax of this command is:-
- **DROP TABLE <table_name>/DROP <database name>;**

DROP TABLE

- **DROP TABLE <Table Name>**

- **Examples:**
- **DROP TABLE Employee;**
- **DROP TABLE Student;**

(This command will remove the table student from the database)

TRUNCATE

- A truncate SQL statement is used to remove all rows (complete data) from a table. It is similar to the DELETE statement with no WHERE clause.
- Drop table command can also be used to delete complete table but it deletes table structure too. TRUNCATE TABLE doesn't delete the structure of the table.
- Truncate table is faster and uses lesser resources than DELETE TABLE command.

TRUNCATE TABLE

- **TRUNCATE TABLE <Table Name>**

- **Examples:**
- **TRUNCATE TABLE Employee;**
- **TRUNCATE TABLE Student;**

(This command will remove all the values from student table)