

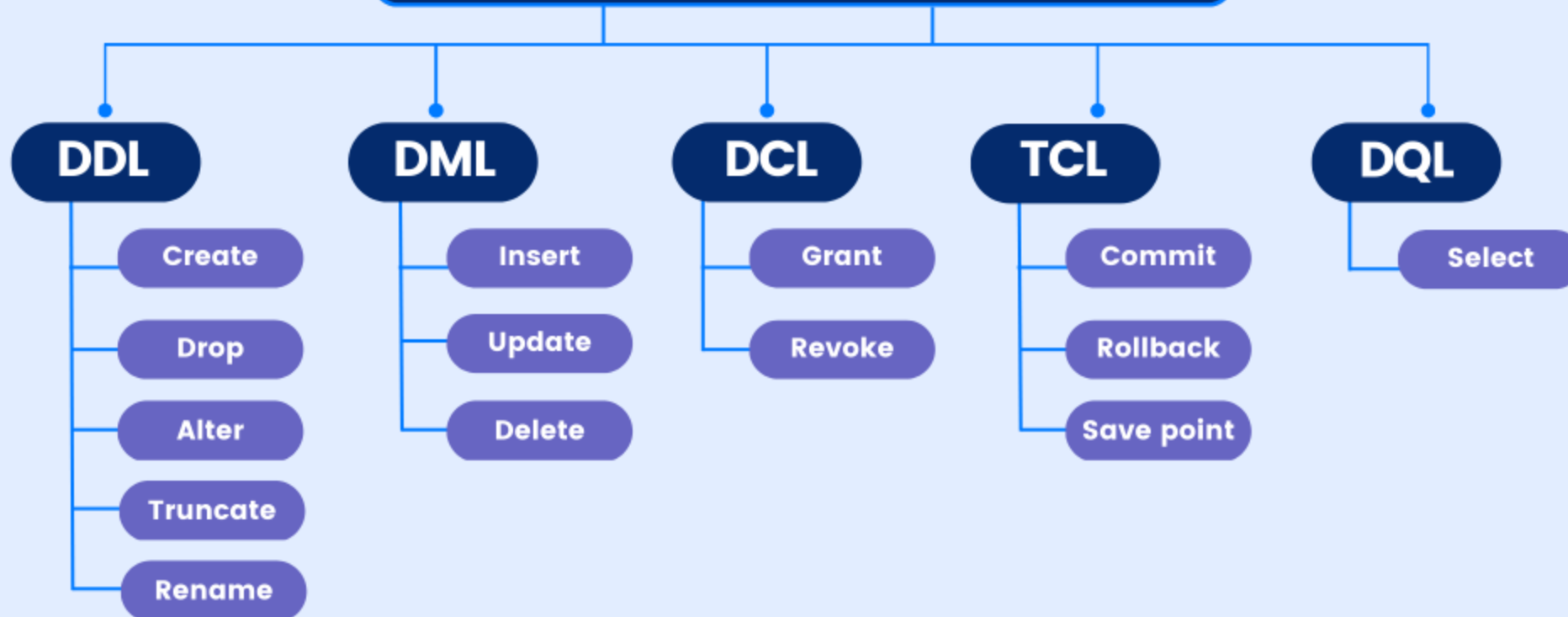


DDL Commands and SQL Datatypes

Agenda

- SQL Commands
 - DDL
 - DML
 - TCL
 - DQL
 - DCL
- SQL Datatypes
 - Numeric
 - Date
 - Datetime
 - Timestamp
 - String

SQL Commands



DDL – Data Definition Language



Data Definition Language actually consists of the SQL commands that can be used to define the database schema.



It simply deals with descriptions of the database schema and is used to create and modify the structure of database objects in the database.



Let's understand all the keywords in DDL.

CREATE

- Create keyword can be used to create different database objects like tables, indexes, views etc.
- A table in databases is used to store data in a tabular format i.e. in form of rows and columns. It is similar like spreadsheet.
- A table creation command requires three things i.e. Name of Table, Name of fields (columns) in table, Data Type for each column.
 - Syntax:
 - `CREATE TABLE TABLE_NAME (
COLUMN1 DATATYPE,
COLUMN2 DATATYPE);`



DML & TCL

- Data Manipulation Language helps us to work on the exact data points i.e. it helps us to work with the rows in a table as rows in a table consist of exact data points.
- Whenever, you make changes to the data in the table you need to confirm that whether you would like to save the changes that you have made or you would like to undo those changes.
- For this reason Data Manipulation Language and Transaction Control Language goes hand in hand as DML will help you to make changes to the data values and TCL will help you to save or undo the changes that you have made.
- Every time you write a DML statement a transaction gets initiated in the background and you need to write TCL statements to close that transaction.



DQL (Data Query Language)

- DQL consist of one keyword called as **SELECT**.
- Whenever we want to query the data from the table in a database we will make use of the SELECT statement.
- If you would like to see all the data in a table in a database you can write the below query (* represents all) which means you will be able to see all the rows and columns in your data.
- If you want to see only limited columns you need to write all the column names separated by comma in front of select statement.

Synatx :

Select * from table_name;

Select col1, col2,col3 from table_name;

DCL (Data Control Language)

- DCL languages are used to control the user access to database, tables and other objects in the databases.
- DCL consist of two keywords
 - GRANT – Whenever, we need to provide access to the user for database or its objects we make use of GRANT statement.
 - REVOKE – Whenever, we need to withdraw the access of a user from the database or its objects we make use of REVOKE statement.

MySQL Data Types

- MySQL supports a wide variety of data types. Some of the most commonly used data types are listed below.
 - NUMERIC
 - DATE
 - DATETIME
 - TIMESTAMP
 - STRING

Type	Length in Bytes	Minimum Value (Signed)	Maximum Value (Signed)	Minimum Value (Unsigned)	Maximum Value (Unsigned)
TINYINT	1	-128	127	0	255
SMALLINT	2	-32768	32767	0	65535
MEDIUMINT	3	-8388608	8388607 to	0	16777215
INT	4	-2147483648	2147483647	0	4294967295
BIGINT	8	-9223372036854775808	92233720368 54775807	0	184467440737 09551615

INTEGER TYPES

Type	Length in Bytes	Minimum Value (Signed)	Maximum Value (Signed)	Minimum Value (Unsigned)	Maximum Value (Unsigned)
FLOAT	4	-3.402823466E+38	-1.175494351E-38	1.175494351E-38	3.402823466E+38
DOUBLE	8	-1.7976931348623 157E+ 308	-2.22507385850720 14E- 308	0, and 2.22507385850720 14E- 308	1.797693134862315 7E+ 308

FLOATING-POINT TYPES

Fixed – Point Types

- There is another data type as DECIMAL which is written as DECIMAL (10,3) where 10 is called as the precision and 3 is the scale. It means we will be able to store any value with 10 digits and 3 decimals.
- For monetary or currency type data use datatype as Decimal.
- Eg. $\text{Rs.}10/3 = 3.3333\dots$
- If we convert back $3.3333\dots$ to 10 some loss of amount will be there (9.9999). You will not get 10 back. If many such values you are adding, there will be big loss of money.
- The maximum number of digits for DECIMAL is 65, but the actual range for a given DECIMAL column can be constrained by the precision or scale of a given column.

DATETIME, DATE and TIMESTAMP

Types	Description	Display Format	Range
DATETIME	Use when you need values containing both date and time information.	YYYY-MM-DD HH:MM:SS	'1000-01-01 00:00:00' to '9999-12-31 23:59:59'.
DATE	Use when you need only date information.	YYYY-MM-DD	'1000-01-01' to '9999-12-31'.
TIMESTAMP	Values are converted from the current timezone to UTC while storing, and converted back from UTC to the current time zone when retrieved.	YYYY-MM-DD HH:MM:SS	'1970-01-01 00:00:01' UTC to '2038-01-19 03:14:07' UTC

TIME TYPE

- ▶ MySQL fetches and displays TIME values in 'HH:MM:SS' format or 'HHH:MM:SS' format
The range of TIME values from '-838:59:59' to '838:59:59'. The hours part may be rather large because not only the TIME type can be used to represent the time of day, i.e. less than 24 hours, but also the passed time or a time of interval between two events.
- ▶ The TIME values in MySQL can be recognized in different formats, some of which can include a trailing fractional seconds part in up to 6 digits microseconds precision. The range for TIME values is '-838:59:59.000000' to '838:59:59.000000'.

YEAR TYPE

► The *YEAR* type is a 1-byte type used to represent year values. It can be declared as *YEAR(2)* or *YEAR(4)* to specify a display width of two or four characters. If no width is given the default is four characters

String length	Range
4-digit string	'1901' to '2155'.
4-digit number	1901 to 2155.
1- or 2-digit string	'0' to '99'. Values in the ranges '0' to '69' and '70' to '99' are converted to YEAR values in the ranges 2000 to 2069 and 1970 to 1999.
1- or 2-digit number	1 to 99. Values in the ranges 1 to 69 and 70 to 99 are converted to YEAR values in the ranges 2001 to 2069 and 1970 to 1999.



STRING TYPES

CHAR and VARCHAR Types

► *The CHAR and VARCHAR types are similar, but differ in the way they are stored and retrieved. They also differ in maximum length and in whether trailing spaces are retained.*

Types	Description	Display Format	Range in characters
CHAR	Contains non-binary strings. Length is fixed as you declare while creating a table. When stored, they are right-padded with spaces to the specified length.	Trailing spaces are removed.	The length can be any value from 0 to 255.
VARCHAR	Contains non-binary strings. Columns are variable-length strings.	As stored.	A value from 0 to 255 before MySQL 5.0.3, and 0 to 65,535 in 5.0.3 and later versions.

Thank You!