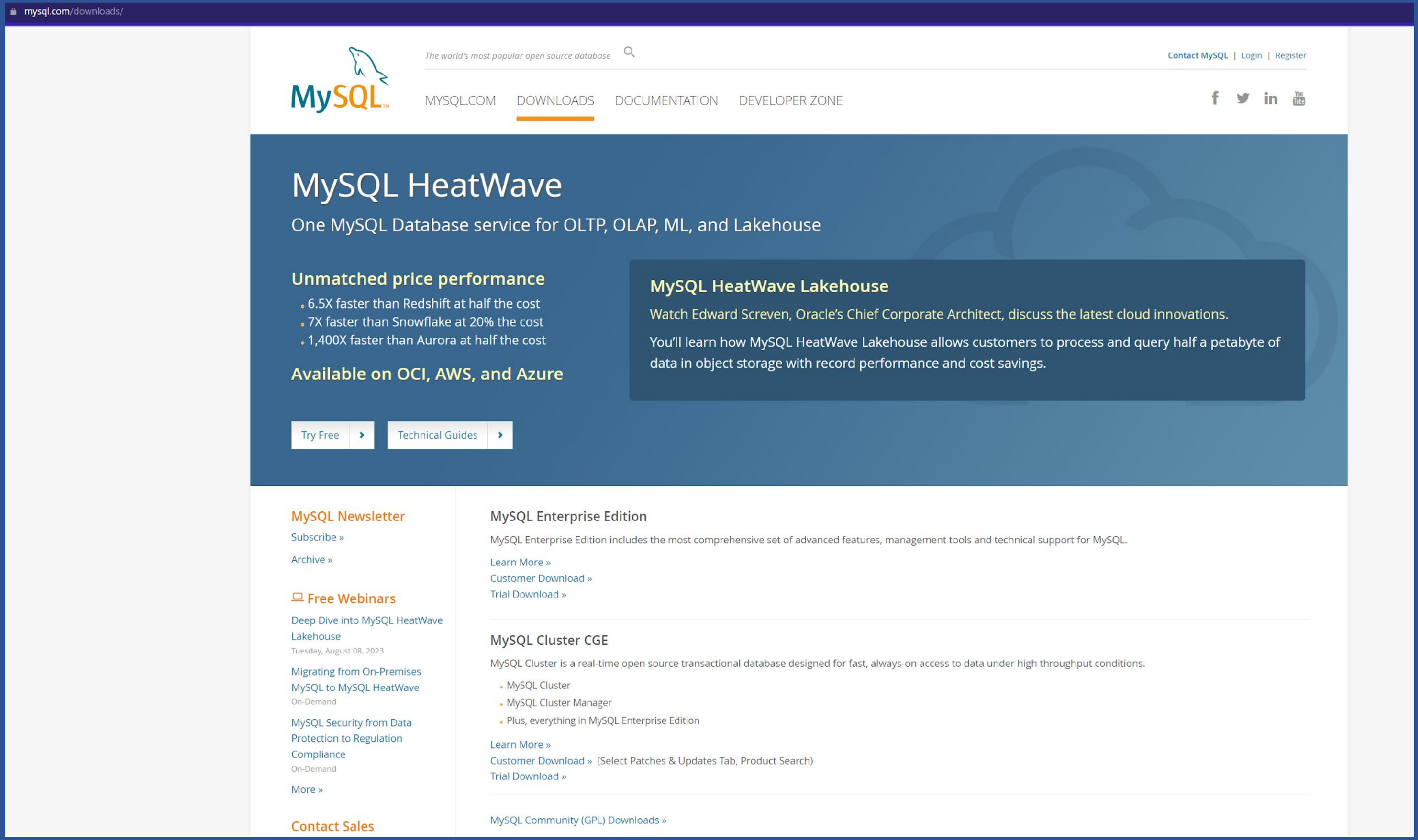
Installing MySQL

Grow Data Skills

This is the site to install MySQL- https://www.mysql.com/downloads/



mysql.com/downloads/

The world's most popular open source database
Contact MySQL | Login | Register
MYSQL.COM
DOWNLOADS
DOCUMENTATION
DEVELOPER ZONE

MySQL HeatWave

One MySQL Database service for OLTP, OLAP, ML, and Lakehouse

Unmatched price performance

- 6.5X faster than Redshift at half the cost
- 7X faster than Snowflake at 20% the cost
- 1,400X faster than Aurora at half the cost

Available on OCI, AWS, and Azure

MySQL HeatWave Lakehouse

Watch Edward Screven, Oracle's Chief Corporate Architect, discuss the latest cloud innovations.

You'll learn how MySQL HeatWave Lakehouse allows customers to process and query half a petabyte of data in object storage with record performance and cost savings.

MySQL Newsletter

Subscribe »

Archive »

Free Webinars

Deep Dive into MySQL HeatWave Lakehouse
Tuesday, August 08, 2023

Migrating from On-Premises MySQL to MySQL HeatWave
On-Demand

MySQL Security from Data Protection to Regulation Compliance
On-Demand

More »

Contact Sales

MySQL Enterprise Edition

MySQL Enterprise Edition includes the most comprehensive set of advanced features, management tools and technical support for MySQL.

Learn More »
Customer Download »
Trial Download »

MySQL Cluster CGE

MySQL Cluster is a real-time open source transactional database designed for fast, always-on access to data under high throughput conditions.

- MySQL Cluster
- MySQL Cluster Manager
- Plus, everything in MySQL Enterprise Edition

Learn More »
Customer Download » (Select Patches & Updates Tab, Product Search)
Trial Download »

MySQL Community (GPL) Downloads »

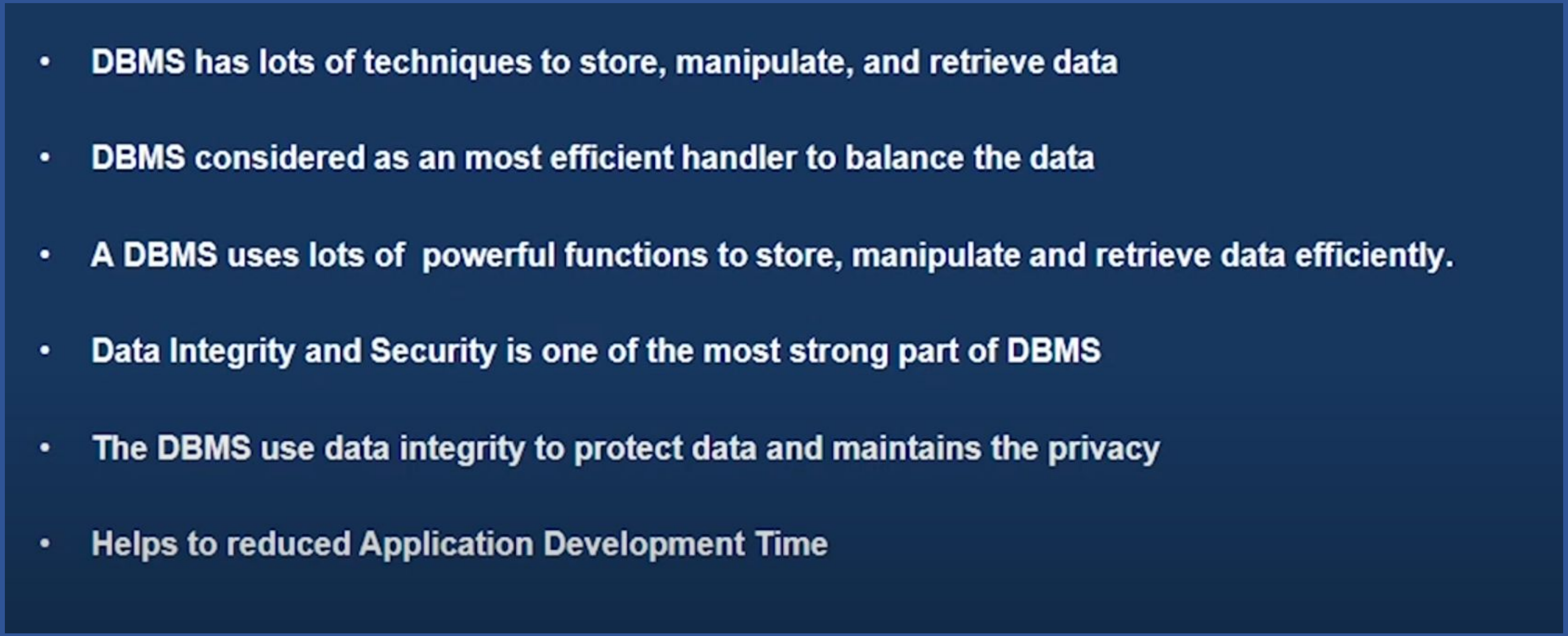
Database Management System(DBMS)

A Database Management System (DBMS) is basically a software where you can store, retrieve, define, and manage your data in a database.

Most used DBMS

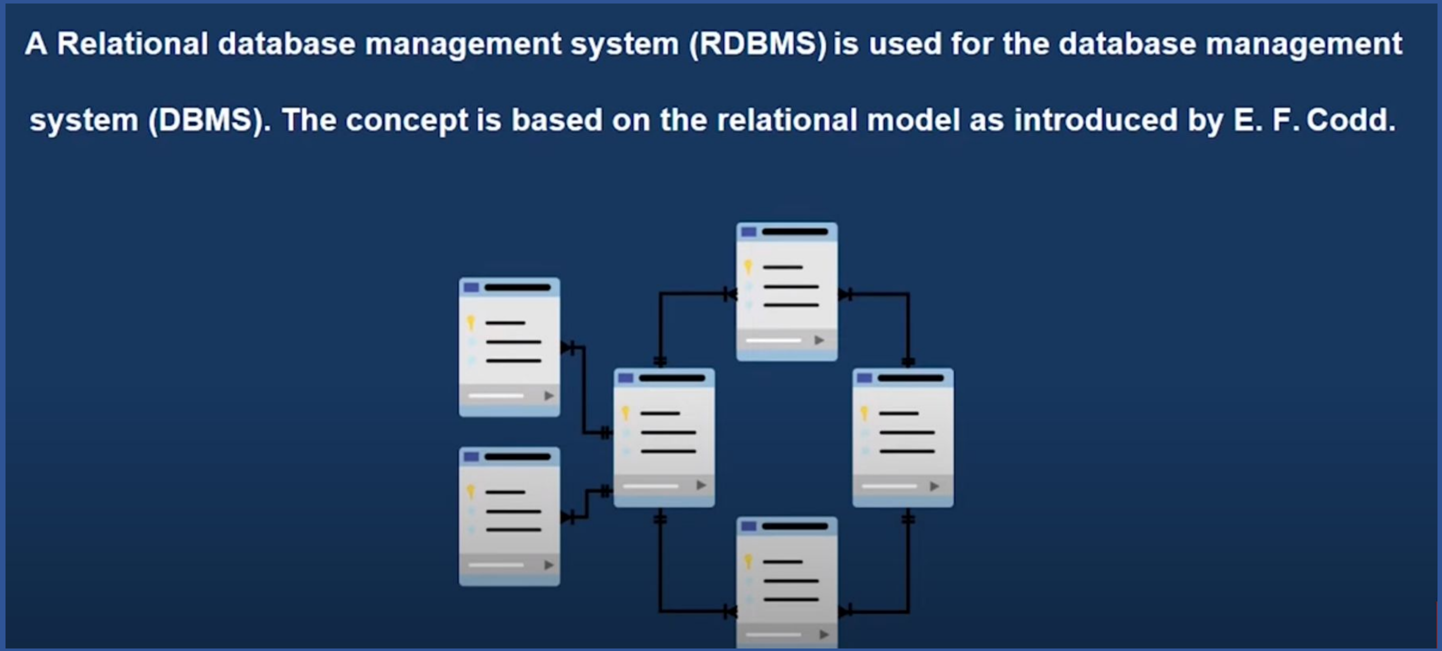
- MySQL
- Oracle
- Microsoft SQL
- PostgreSQL
- MongoDB

Advantages of DBMS



- DBMS has lots of techniques to store, manipulate, and retrieve data
- DBMS considered as an most efficient handler to balance the data
- A DBMS uses lots of powerful functions to store, manipulate and retrieve data efficiently.
- Data Integrity and Security is one of the most strong part of DBMS
- The DBMS use data integrity to protect data and maintains the privacy
- Helps to reduced Application Development Time

RDBMS



A Relational database management system (RDBMS) is used for the database management system (DBMS). The concept is based on the relational model as introduced by E. F. Codd.

- The data in RDBMS is stored in database objects called tables. A table is a collection of related data entries, and it consists of columns and rows.
- A record, also called a row, is each individual entry that exists in a table.
- A column is a vertical entity in a table that contains all information associated with a specific field in a table.

What is SQL?

- SQL stands for Structured Query Language
- SQL lets you access and manipulate databases
- SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization (ISO) in 1987

What Can SQL do?

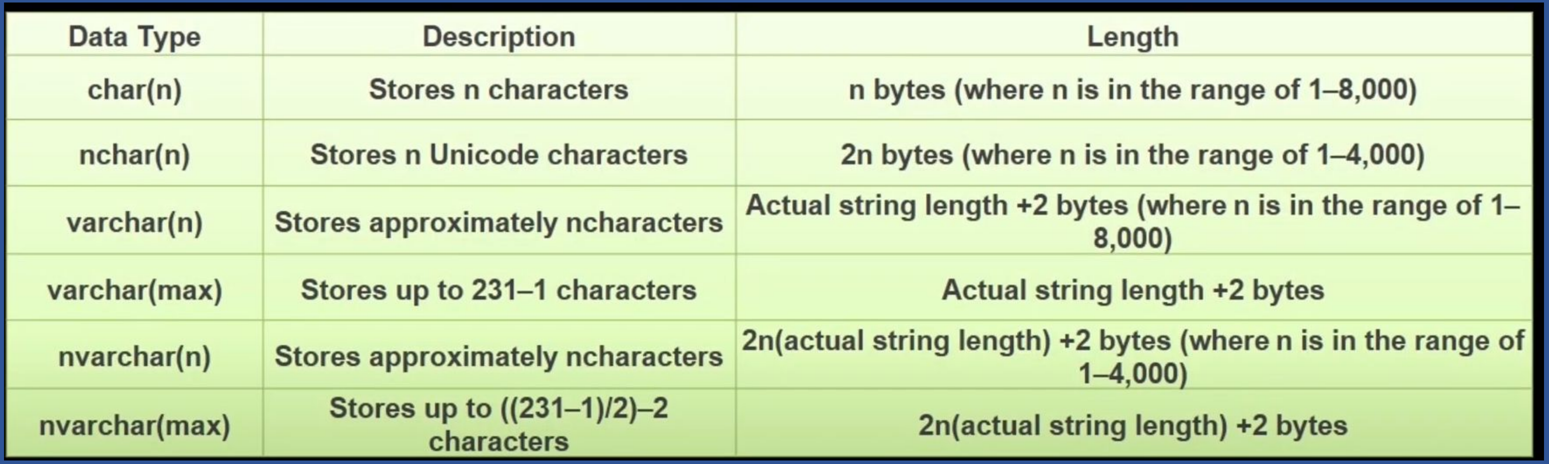
- SQL can execute queries against a database
- SQL can retrieve data from a database
- SQL can insert records in a database
- SQL can update records in a database
- SQL can delete records from a database
- SQL can create new databases
- SQL can create new tables in a database
- SQL can create stored procedures in a database
- SQL can create views in a database
- SQL can set permissions on tables, procedures, and views

SQL is very powerful

MySQL Data Type

The data type of a column defines what value the column can hold: integer, character, money, date and time, binary, and so on.
In MySQL there are three main data types: string, numeric, and date and time.

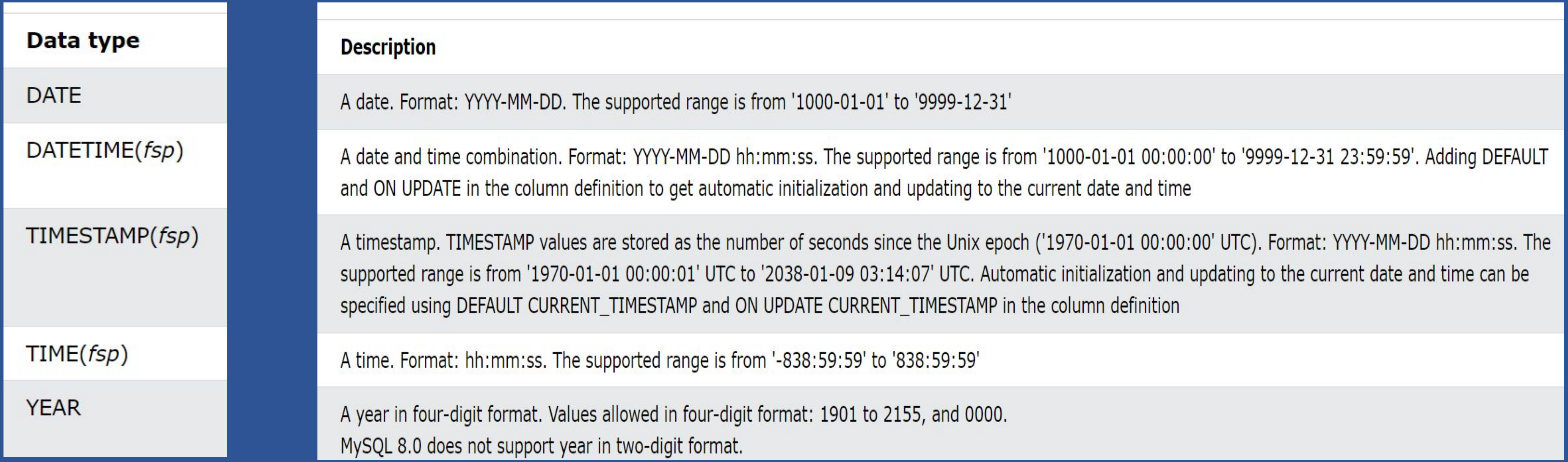
String Data types in MySQL



Data Type	Description	Length
char(n)	Stores n characters	n bytes (where n is in the range of 1–8,000)
nchar(n)	Stores n Unicode characters	2n bytes (where n is in the range of 1–4,000)
varchar(n)	Stores approximately ncharacters	Actual string length +2 bytes (where n is in the range of 1–8,000)
varchar(max)	Stores up to 231–1 characters	Actual string length +2 bytes
nvarchar(n)	Stores approximately ncharacters	2n(actual string length) +2 bytes (where n is in the range of 1–4,000)
nvarchar(max)	Stores up to ((231–1)/2)–2 characters	2n(actual string length) +2 bytes

MySQL Data Type

Date Data types in MySQL

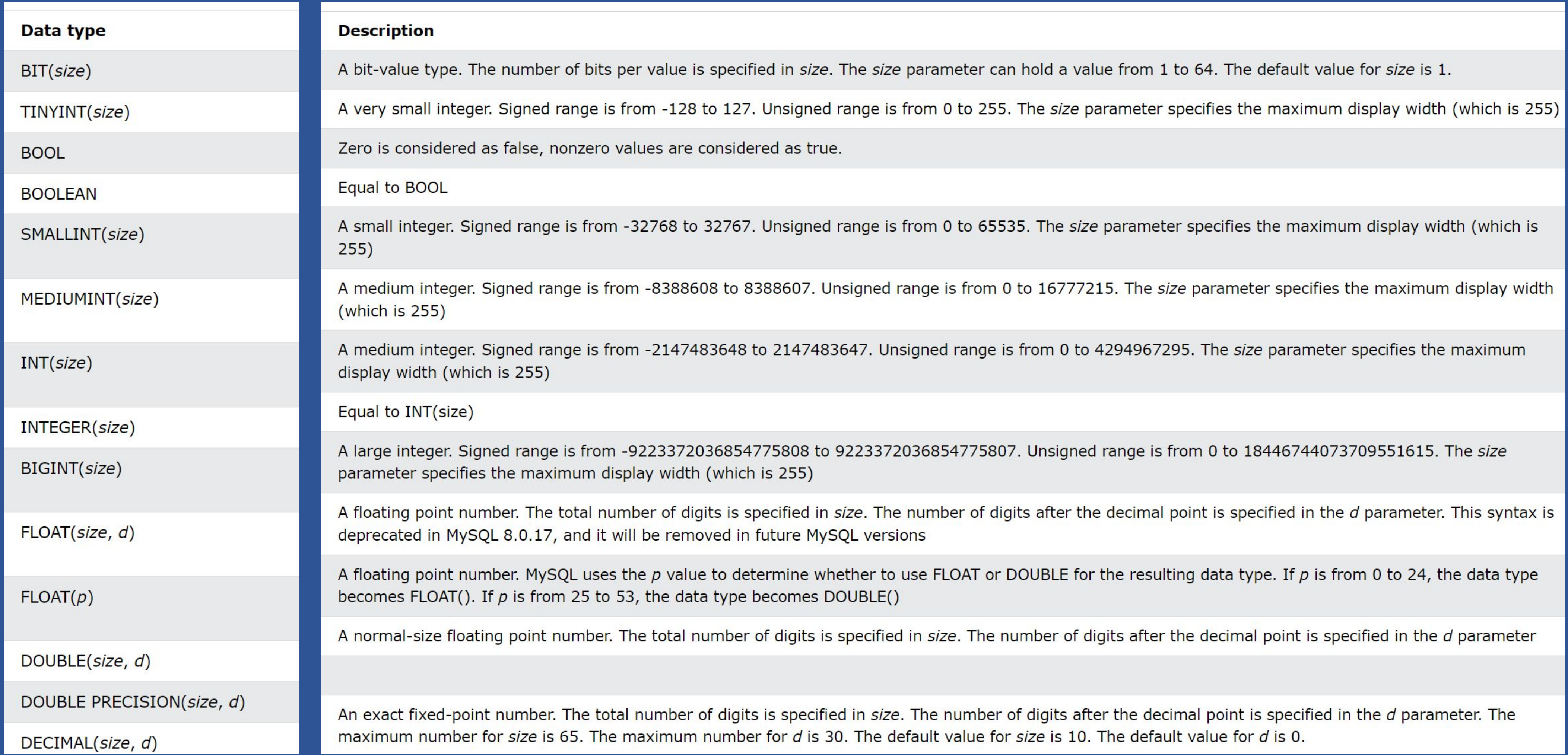


Data type	Description
DATE	A date. Format: YYYY-MM-DD. The supported range is from '1000-01-01' to '9999-12-31'
DATETIME(*fsp*)	A date and time combination. Format: YYYY-MM-DD hh:mm:ss. The supported range is from '1000-01-01 00:00:00' to '9999-12-31 23:59:59'. Adding DEFAULT and ON UPDATE in the column definition to get automatic initialization and updating to the current date and time
TIMESTAMP(*fsp*)	A timestamp. TIMESTAMP values are stored as the number of seconds since the Unix epoch ('1970-01-01 00:00:00' UTC). Format: YYYY-MM-DD hh:mm:ss. The supported range is from '1970-01-01 00:00:01' UTC to '2038-01-09 03:14:07' UTC. Automatic initialization and updating to the current date and time can be specified using DEFAULT CURRENT_TIMESTAMP and ON UPDATE CURRENT_TIMESTAMP in the column definition
TIME(*fsp*)	A time. Format: hh:mm:ss. The supported range is from '-838:59:59' to '838:59:59'
YEAR	A year in four-digit format. Values allowed in four-digit format: 1901 to 2155, and 0000. MySQL 8.0 does not support year in two-digit format.

Grow Data Skills

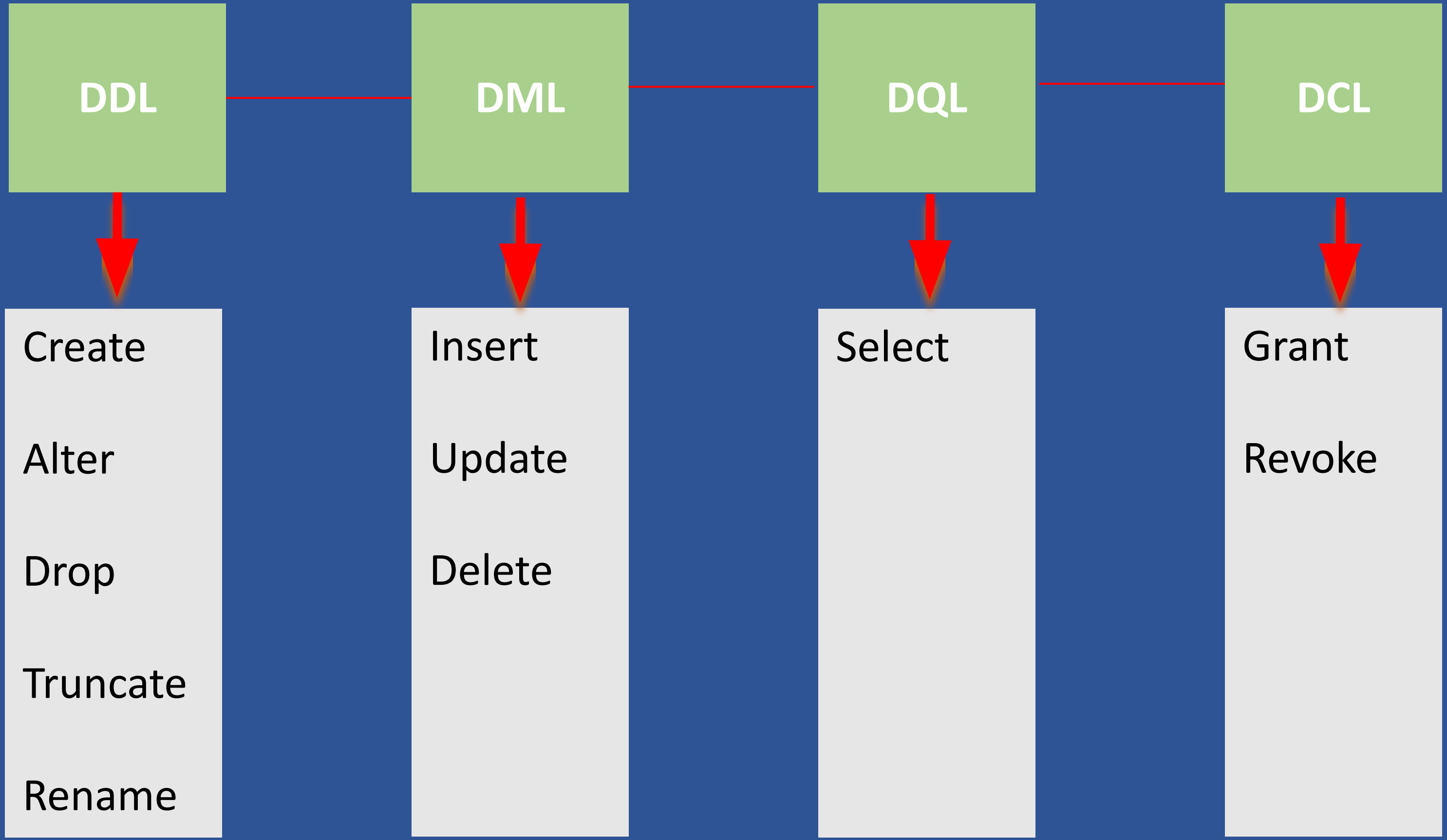
MySQL Data Type

Numeric Data types in MySQL



Data type	Description
BIT(*size*)	A bit-value type. The number of bits per value is specified in *size*. The *size* parameter can hold a value from 1 to 64. The default value for *size* is 1.
TINYINT(*size*)	A very small integer. Signed range is from -128 to 127. Unsigned range is from 0 to 255. The *size* parameter specifies the maximum display width (which is 255)
BOOL	Zero is considered as false, nonzero values are considered as true.
BOOLEAN	Equal to BOOL
SMALLINT(*size*)	A small integer. Signed range is from -32768 to 32767. Unsigned range is from 0 to 65535. The *size* parameter specifies the maximum display width (which is 255)
MEDIUMINT(*size*)	A medium integer. Signed range is from -8388608 to 8388607. Unsigned range is from 0 to 16777215. The *size* parameter specifies the maximum display width (which is 255)
INT(*size*)	A medium integer. Signed range is from -2147483648 to 2147483647. Unsigned range is from 0 to 4294967295. The *size* parameter specifies the maximum display width (which is 255)
INTEGER(*size*)	Equal to INT(size)
BIGINT(*size*)	A large integer. Signed range is from -9223372036854775808 to 9223372036854775807. Unsigned range is from 0 to 18446744073709551615. The *size* parameter specifies the maximum display width (which is 255)
FLOAT(*size*, *d*)	A floating point number. The total number of digits is specified in *size*. The number of digits after the decimal point is specified in the *d* parameter. This syntax is deprecated in MySQL 8.0.17, and it will be removed in future MySQL versions
FLOAT(*p*)	A floating point number. MySQL uses the *p* value to determine whether to use FLOAT or DOUBLE for the resulting data type. If *p* is from 0 to 24, the data type becomes FLOAT(). If *p* is from 25 to 53, the data type becomes DOUBLE()
DOUBLE(*size*, *d*)	A normal-size floating point number. The total number of digits is specified in *size*. The number of digits after the decimal point is specified in the *d* parameter
DOUBLE PRECISION(*size*, *d*)	
DECIMAL(*size*, *d*)	An exact fixed-point number. The total number of digits is specified in *size*. The number of digits after the decimal point is specified in the *d* parameter. The maximum number for *size* is 65. The maximum number for *d* is 30. The default value for *size* is 10. The default value for *d* is 0.

Types of Commands in SQL



DDL

Create

Alter

Drop

Truncate

Rename

DML

Insert

Update

Delete

DQL

Select

DCL

Grant

Revoke

DDL- Data Definition Language

It consists of SQL commands that can be used to define the database structures but not data.

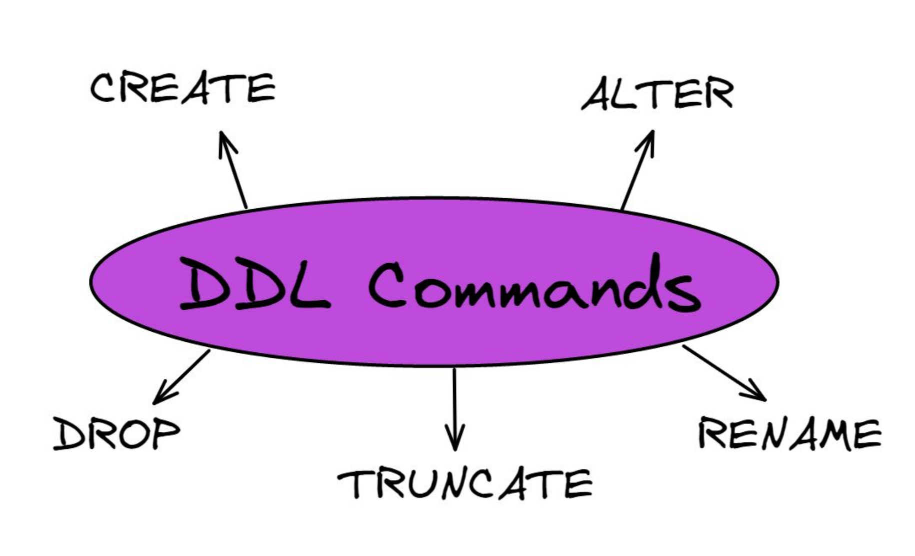
CREATE: This command is used to create the database or its objects (like table, index, function, views, store procedure, and triggers).

DROP: This command is used to delete objects from the database.

ALTER: This is used to alter the structure of the database.

TRUNCATE: This is used to remove all records from a table, including all spaces allocated for the records are removed.

RENAME: This is used to rename an object existing in the database.



SQL CREATE ,DROP, TRUNCATE

- CREATE DATABASE
- DROP DATABASE
- CREATE TABLE
- DROP TABLE
- TRUNCATE TABLE

SQL ALTER TABLE

ALTER TABLE – ADD Column

ALTER TABLE – DROP COLUMN

ALTER TABLE – RENAME Column

ALTER TABLE – ALTER/MODIFY DATATYPE

SQL CONSTRAINTS

SQL constraints are used to specify rules for the data in a table.

Constraints are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the table. If there is any violation between the constraint and the data action, the action is aborted.

Constraints can be column level or table level.

The following constraints are commonly used in SQL:

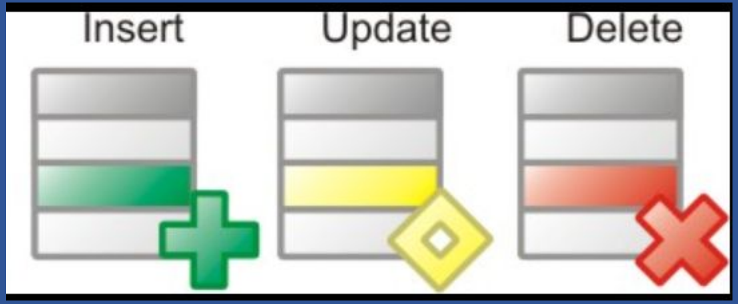
- NOT NULL - Ensures that a column cannot have a NULL value
- UNIQUE - Ensures that all values in a column are different
- PRIMARY KEY - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table
- FOREIGN KEY - Prevents actions that would destroy links between tables
- CHECK - Ensures that the values in a column satisfies a specific condition
- DEFAULT - Sets a default value for a column if no value is specified

DML(Data Manipulation Language)

Grow Data Skills

The SQL commands that deal with the manipulation of data present in the database belong to DML and this includes most of the SQL statements. It is the component of the SQL statement that controls access to data and to the database. Basically, DCL statements are grouped with DML statements.

- **INSERT**: It is used to insert data into a table.
- **UPDATE:** It is used to update existing data within a table.
- **DELETE**: It is used to delete records from a database table.



DQL(Data Query Language)

- It is a SQL statement that allows getting data from the database and imposing order upon it.

- It includes the **SELECT** statement.

- This command allows getting the data out of the database to perform operations with it.

- When a SELECT is fired against a table or tables the result is compiled into a further temporary table, which is displayed or perhaps received by the program i.e. a front-end.

DCL(Data Control Language)

- It includes commands such as GRANT and REVOKE which mainly deal with the rights, permissions, and other controls of the database system.

- **GRANT**: This command gives users access privileges to the database.

- **REVOKE**: This command withdraws the user's access privileges given by using the GRANT command.

Things to remember!!

- SQL keywords are NOT case sensitive:

`select` is the same as `SELECT`

- Some database systems require a semicolon at the end of each SQL statement.
- Semicolon is the standard way to separate each SQL statement in database systems that allow more than one SQL statement to be executed in the same call to the server.
- In this tutorial, we will use semicolon at the end of each SQL statement.