**WEEK 1**

**Introduction to RDBMS(MAIN HEADING)**

A **relational database management system** (**RDBMS**) is a [database management system](https://en.wikipedia.org/wiki/Database_management_system) (DBMS) that is based on the [relational model](https://en.wikipedia.org/wiki/Relational_model) as invented by [E. F. Codd](https://en.wikipedia.org/wiki/Edgar_F._Codd), of IBM's [San Jose Research Laboratory](https://en.wikipedia.org/wiki/IBM_Almaden_Research_Center).

RDBMSs have been a common choice for the storage of information in new databases used for financial records, manufacturing and logistical information, personnel data, and other applications since the 1980s.

RDBMS is the basis for SQL, and for all modern database systems like MS SQL Server, IBM DB2, Oracle, MySQL, and Microsoft Access.

The data in RDBMS is stored in database objects called **tables**. The table is a collection of related data entries and it consists of columns and rows.

RDBMS is based on relational model, in which data is represented in the form of relations, with enforced relationships between the tables.

RDBMS defines the integrity constraint for the purpose of holding ACID PROPERTY.

In RDBMS, normalization process will be present to check the database table consistency

RDBMS helps in recovery of the database in case of loss of data

RDBMS is used to establish the relationship concept between two database objects, i.e, tables

**The Advantages of a Relational Database Management System**

A Relational Database Management System (RDBMS) is a software system that provides access to a relational database. The software system is a collection of software applications that can be used to create, maintain, manage and use the database. A "relational database" is a database structured on the "relational" model. Data are stored and presented in a tabular format, organized in rows and columns with one record per row.

The following are the advantages of RDBMS:

1. DATA STRUCTURE

2. MULTI-USER ACCESS

3. PRIVILEGES

4. NETWORK ACCESS

5. SPEED

6. MAINTENANCE

7. LANGUAGE

**List of**[**relational database management systems**](https://en.wikipedia.org/wiki/Relational_database_management_system):

According to DB-Engines, the most popular systems are Oracle, MySQL, Microsoft SQL Server, PostgreSQL and IBM DB2.

Other RDBMS Softwares are as follows(write any five)

* [4th Dimension](https://en.wikipedia.org/wiki/4th_Dimension_(Software))
* [Adabas D](https://en.wikipedia.org/wiki/Adabas_D)
* [Alpha Five](https://en.wikipedia.org/wiki/Alpha_Five_(database))
* [Apache Derby](https://en.wikipedia.org/wiki/Apache_Derby)
* [Aster Data](https://en.wikipedia.org/wiki/Aster_Data_Systems)
* [Amazon Aurora](https://en.wikipedia.org/wiki/Amazon_Relational_Database_Service)
* [Altibase](https://en.wikipedia.org/wiki/Altibase)
* [CA Datacom](https://en.wikipedia.org/wiki/DATACOM/DB)
* [CA IDMS](https://en.wikipedia.org/wiki/IDMS)
* [Clarion](https://en.wikipedia.org/wiki/Clarion_(programming_language))
* [ClickHouse](https://en.wikipedia.org/wiki/ClickHouse)
* [Clustrix](https://en.wikipedia.org/wiki/Clustrix)
* [CSQL](https://en.wikipedia.org/wiki/CSQL)
* [CUBRID](https://en.wikipedia.org/wiki/CUBRID)
* [DataEase](https://en.wikipedia.org/wiki/DataEase)
* [Derby](https://en.wikipedia.org/wiki/Apache_Derby) aka [Java DB](https://en.wikipedia.org/wiki/Java_DB)
* [Empress Embedded Database](https://en.wikipedia.org/wiki/Empress_database)
* [EXASolution](https://en.wikipedia.org/wiki/EXASOL)

**Applications of RDBMS**

**RDBMS** has its applications in many fields along with the following major fields

1) Inventory

2) Supply chain management

3) Hospitals

4) Schools

5) Library

6) Data Mining and Data Warehouses

**Introduction to SQL**

SQL stands for Structured Query Language

SQL lets you access and manipulate databases

SQL is an ANSI (American National Standards Institute) standard

**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

Although SQL is an ANSI (American National Standards Institute) standard, there are many different versions of the SQL language.

However, to be compliant with the ANSI standard, they all support at least the major commands (such as SELECT, UPDATE, DELETE, INSERT, WHERE) in a similar manner.

# Introduction to MySQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius' daughter, and "SQL", the abbreviation for Structured Query Language.

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation.

MySQL is a central component of the LAMP open-source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python". Applications that use the MySQL database include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, and Drupal. MySQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook, Twitter, Flickr,and YouTube.

**What is MySQL?**

MySQL is a database management system.

MySQL databases are relational.

MySQL software is Open Source.

The MySQL Database Server is very fast, reliable, scalable, and easy to use.

MySQL Server works in client/server or embedded systems.

A large amount of contributed MySQL software is available.

# Top Reasons to Use MySQL

1. Scalability and Flexibility

2. High Performance

3. High Availability

4. Robust Transactional Support

5. Web and Data Warehouse Strengths

6. Strong Data Protection

7. Comprehensive Application Development

8. Management Ease

### 9. Open Source Freedom and 24 x 7 Support

### 10. Lowest Total Cost of Ownership

MySQL Data Types(MAIN HEADING)

In MySQL there are three main types: text, number, and Date/Time types.

**Text types:**

|  |  |
| --- | --- |
| **Data type** | **Description** |
| CHAR(size) | Holds a fixed length string (can contain letters, numbers, and special characters). The fixed size is specified in parenthesis. Can store up to 255 characters |
| VARCHAR(size) | Holds a variable length string (can contain letters, numbers, and special characters). The maximum size is specified in parenthesis. Can store up to 255 characters. **Note:** If you put a greater value than 255 it will be converted to a TEXT type |
| TINYTEXT | Holds a string with a maximum length of 255 characters |
| TEXT | Holds a string with a maximum length of 65,535 characters |
| BLOB | For BLOBs (Binary Large OBjects). Holds up to 65,535 bytes of data |
| MEDIUMTEXT | Holds a string with a maximum length of 16,777,215 characters |
| MEDIUMBLOB | For BLOBs (Binary Large OBjects). Holds up to 16,777,215 bytes of data |
| LONGTEXT | Holds a string with a maximum length of 4,294,967,295 characters |
| LONGBLOB | For BLOBs (Binary Large OBjects). Holds up to 4,294,967,295 bytes of data |
| ENUM(x,y,z,etc.) | Let you enter a list of possible values. You can list up to 65535 values in an ENUM list. If a value is inserted that is not in the list, a blank value will be inserted.  **Note:** The values are sorted in the order you enter them.  You enter the possible values in this format: ENUM('X','Y','Z') |
| SET | Similar to ENUM except that SET may contain up to 64 list items and can store more than one choice |

**Number types:**

|  |  |
| --- | --- |
| **Data type** | **Description** |
| TINYINT(size) | -128 to 127 normal. 0 to 255 UNSIGNED\*. The maximum number of digits may be specified in parenthesis |
| SMALLINT(size) | -32768 to 32767 normal. 0 to 65535 UNSIGNED\*. The maximum number of digits may be specified in parenthesis |
| MEDIUMINT(size) | -8388608 to 8388607 normal. 0 to 16777215 UNSIGNED\*. The maximum number of digits may be specified in parenthesis |
| INT(size) | -2147483648 to 2147483647 normal. 0 to 4294967295 UNSIGNED\*. The maximum number of digits may be specified in parenthesis |
| BIGINT(size) | -9223372036854775808 to 9223372036854775807 normal. 0 to 18446744073709551615 UNSIGNED\*. The maximum number of digits may be specified in parenthesis |
| FLOAT(size,d) | A small number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter |
| DOUBLE(size,d) | A large number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter |
| DECIMAL(size,d) | A DOUBLE stored as a string , allowing for a fixed decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter |

\*The integer types have an extra option called UNSIGNED. Normally, the integer goes from an negative to positive value. Adding the UNSIGNED attribute will move that range up so it starts at zero instead of a negative number.

**Date types:**

|  |  |
| --- | --- |
| **Data type** | **Description** |
| DATE() | A date. Format: YYYY-MM-DD  **Note:** The supported range is from '1000-01-01' to '9999-12-31' |
| DATETIME() | \*A date and time combination. Format: YYYY-MM-DD HH:MM:SS  **Note:** The supported range is from '1000-01-01 00:00:00' to '9999-12-31 23:59:59' |
| TIMESTAMP() | \*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  **Note:** The supported range is from '1970-01-01 00:00:01' UTC to '2038-01-09 03:14:07' UTC |
| TIME() | A time. Format: HH:MM:SS  **Note:** The supported range is from '-838:59:59' to '838:59:59' |
| YEAR() | A year in two-digit or four-digit format.  **Note:** Values allowed in four-digit format: 1901 to 2155. Values allowed in two-digit format: 70 to 69, representing years from 1970 to 2069 |

\*Even if DATETIME and TIMESTAMP return the same format, they work very differently. In an INSERT or UPDATE query, the TIMESTAMP automatically set itself to the current date and time. TIMESTAMP also accepts various formats, like YYYYMMDDHHMMSS, YYMMDDHHMMSS, YYYYMMDD, or YYMMDD.

**SQL Commands:**

**SQL commands** are instructions, coded into SQL statements, which are used to communicate with the database to perform specific tasks, work, functions and queries with data.

SQL commands can be used not only for searching the database but also to perform various other functions like, for example, you can [*create tables*](http://beginner-sql-tutorial.com/sql-create-statement.htm), add data to tables, or modify data, drop the table, set permissions for users. SQL commands are grouped into four major categories depending on their functionality:

* **Data Definition Language (DDL)** - These SQL commands are used for creating, modifying, and dropping the structure of database objects.

The commands are

CREATE DATABASE - creates a new database

ALTER DATABASE - modifies a database

CREATE TABLE - creates a new table

ALTER TABLE - modifies a table

DROP TABLE - deletes a table

CREATE INDEX - creates an index (search key)

DROP INDEX - deletes an index

* **Data Manipulation Language (DML)** - These SQL commands are used for storing, retrieving, modifying, and deleting data.

These Data Manipulation Language commands are:

* SELECT - extracts data from a database
* UPDATE - updates data in a database
* DELETE - deletes data from a database
* INSERT INTO - inserts new data into a database
* **Transaction Control Language (TCL)** - These SQL commands are used for managing changes affecting the data.

These commands are

* COMMIT-to save the changes.
* ROLLBACK-to roll back the changes.
* SAVEPOINT-creates points within groups of transactions in which to ROLLBACK
* **Data Control Language (DCL)** - These SQL commands are used for providing security to database objects.

These commands are

* GRANT-used to provide access or privileges on the database objects to the users.
* REVOKE-removes user access rights or privileges to the database objects.