

Databases

ashok.dudhat@devugees.org

What is Data?

In simple words **data** can be facts related to any object in consideration.

For example your name, age, height, weight, etc are some data related to you.

A picture , image , file , pdf etc can also be considered data.

What is Database?

Database is a systematic collection of data. Databases support storage and manipulation of data. Databases make data management easy.

Let's discuss **few examples**.

An **online telephone directory** would definitely use database to store data pertaining to people, phone numbers, other contact details, etc.

Your **electricity service provider** is obviously using a database to manage billing , client related issues, to handle fault data, etc.

Let's also consider the **facebook**. It needs to store, manipulate and present data related to members, their friends, member activities, messages, advertisements and lot more.

What is a Database Management System (DBMS)?

Database Management System (DBMS) is a collection of programs which enables its users to access **database**, manipulate **data**, reporting / representation of **data** .

It also helps to control access to the **database**.

Types of DBMS

There are 4 major types of DBMS.

- Hierarchical
- Network DBMS
- Relational DBMS
- Object Oriented Relation DBMS

Types of DBMS

- **Hierarchical** - this type of DBMS employs the "**parent-child**" relationship of storing data. This type of DBMS is **rarely used nowadays**. Its structure is like a tree with nodes representing records and branches representing fields.

Example : The windows registry used in Windows XP. Configuration settings are stored as tree structures with nodes.

- **Network DBMS** - this type of DBMS **supports many-to many relations**. This usually results in complex database structures.

Example: RDM Server is an example of a database management system that implements the network model.

Types of DBMS

- **Relational DBMS** - this type of DBMS defines database relationships in form of tables, also known as relations. Unlike network DBMS, RDBMS **does not support many to many relationships**. Relational DBMS usually have **pre-defined data types that they can support**. This is the most popular DBMS type in the market.

Examples : MySQL, Oracle, and Microsoft SQL Server database.

- **Object Oriented Relation DBMS** - this type supports storage of new data types. **The data to be stored is in form of objects**. The objects to be stored in the database have attributes (i.e. gender, age) and methods that define what to do with the data.

Examples: PostgreSQL

What is SQL?

Structured Query language (SQL) **pronounced as "S-Q-L" or sometimes as "See-Quel"** is actually the standard language for dealing with Relational Databases.

SQL programming can be effectively used to insert, search, update, delete database records.

That doesn't mean SQL cannot do things beyond that. In fact it can do lot of things including, but not limited to, optimizing and maintenance of databases.

Relational databases like MySQL Database, Oracle, Ms SQL server, Sybase, etc uses **SQL** !

How to use sql syntaxes?

SQL syntaxes used in these databases are almost similar, except the fact that some are using few different syntaxes and even proprietary SQL syntaxes.

SQL Example

```
SELECT * FROM Members WHERE Age > 30
```

RDBMS Terminology

Before we proceed to explain the MySQL database system, let us revise a few definitions related to the database.

- **Database** – A database is a collection of tables, with related data.
- **Table** – A table is a matrix with data. A table in a database looks like a simple spreadsheet.
- **Column** – One column (data element) contains data of one and the same kind, for example the column postcode.
- **Row** – A row (= tuple, entry or record) is a group of related data, for example the data of one subscription.
- **Redundancy** – Storing data twice, redundantly to make the system faster.
- **Primary Key** – A primary key is unique. A key value can not occur twice in one table. With a key, you can only find one row.
- **Foreign Key** – A foreign key is the linking pin between two tables.
- **Compound Key** – A compound key (composite key) is a key that consists of multiple columns, because one column is not sufficiently unique.
- **Index** – An index in a database resembles an index at the back of a book.
- **Referential Integrity** – Referential Integrity makes sure that a foreign key value always points to an existing row.

What is MySQL?

MySQL is a freely available open source Relational Database Management System (RDBMS) that uses Structured Query Language (**SQL**).

MySQL Server Installation

Install the MySQL server by using the Ubuntu package manager:

```
sudo apt-get update  
sudo apt-get install mysql-server
```

Allow remote access

```
sudo ufw allow mysql
```

Start the MySQL service

```
systemctl start mysql
```

Start the mysql shell

There is more than one way to work with a MySQL server, but we will go with most basic and compatible approach, the **mysql shell**.

At the command prompt, run the following command to launch the the **mysql shell** and enter it as the root user:

```
/usr/bin/mysql -u root -p
```

When you're prompted for a password, enter the one that you set at installation time, or if you haven't set one, press **Enter** to submit no password.

The following **mysql shell prompt** should appear:

```
mysql>
```

Set the root password

If you logged in by entering a blank password, or if you want to change the root password that you set, you can create or change the password.

Enter the following command in the mysql shell, replace password with your new password:

```
UPDATE mysql.user SET Password = PASSWORD('password') WHERE  
User = 'root';
```

To make the change take effect, reload the stored user information with the following command:

```
FLUSH PRIVILEGES;
```

Remove Old MySQL Server

Run the following command :

```
sudo apt autoremove mysql-server
```

Installation MySQL Workbench

Before You Begin[Permalink](#)

Update repositories and upgrade if necessary:

```
sudo apt update && sudo apt upgrade
```

Install MySQL Workbench

Install MySQL Workbench using the APT package manager:

```
sudo apt install mysql-workbench
```

Run MySQL Workbench

Launch MySQL Workbench from the terminal:

```
mysql-workbench
```