



Introduction to MySQL

Agenda

- Database Concepts
- Why SQL
- What is DBMS?
- What is RDBMS?
- Types of DBMS

What is database?

- Collection of related data or tables.
- Table has rows and columns.
- Data can be Structured or Unstructured.
- Eg. Students (sid,sname,sage,course), Department (depid,dname,location)

Why Databases?

1. Handling Large Datasets

- **MySQL:** MySQL is designed to handle large datasets with millions or even billions of rows efficiently. Databases like MySQL are optimized for querying, indexing, and retrieving data quickly, even in massive tables.
- **Excel:** Excel has limitations when dealing with large datasets. Although Excel can handle up to about 1 million rows (as of Excel 2016 and later), it starts to slow down and become cumbersome when dealing with very large datasets or complex calculations. (Approximately 255 sheets, one worksheet has 1,048,576 rows and 16,384 columns).

2. Multi-User Access and Concurrency

- **MySQL:** MySQL is a multi-user database system that allows multiple users to read and write data simultaneously or at the same time.
- **Excel:** Excel is primarily a single-user tool. While Excel does support shared workbooks and collaboration features, concurrent edits by multiple users can lead to conflicts and data integrity issues, especially with complex or large spreadsheets.

3. Security and User Permissions

- **MySQL:** MySQL offers robust security features, including user authentication, role-based access control, and encryption. You can define specific permissions for different users, ensuring that only authorized users can view, modify, or delete data.
- **Excel:** Excel has basic password protection and file encryption, but it lacks fine-grained security controls.

4. Automation and Integration

- **MySQL:** You can use MySQL with a Python script to automatically generate daily reports, update records, or synchronize data between systems.
- **Excel:** It requires manual intervention for similar tasks unless VBA macros are used, which can be more limited and less scalable.

- **When to Use MySQL vs. Excel:**

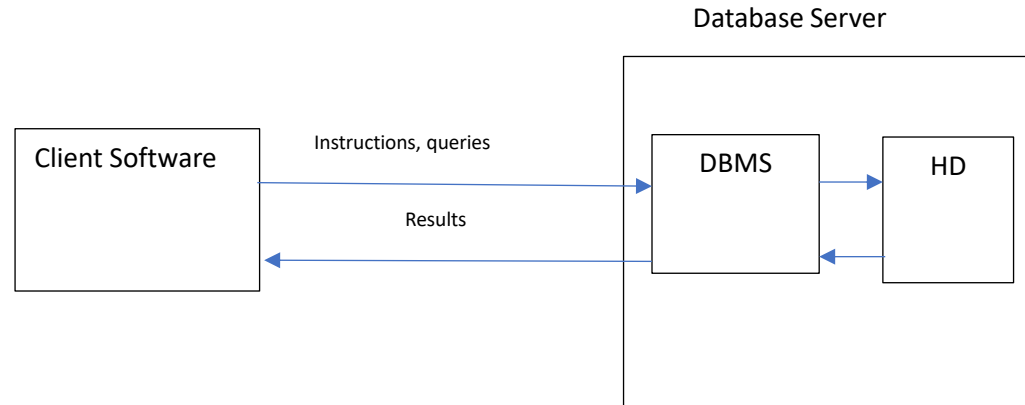
- **Use MySQL when:**

- You need to handle large datasets with complex relationships.
- Multiple users need to access and update the data concurrently.
- Data integrity, security, and backup are critical.
- You require automation, advanced querying, and integration with other systems.
- You need to scale your application or data over time.

- **Use Excel when:**

- You are working with smaller datasets that fit within Excel's limits.
- You need to perform quick calculations or simple data manipulation.
- Your project doesn't require multi-user access or complex data relationships.

- Any file you create is saved on your hard disk by default.
- To save, access or update the file, there is a mediator in between **you** and **hard disk**. It is called an **operating system**.
- Similarly, between **User** and **Hard Disk** there is a mediator called Database Management System (**DBMS**).



- **DBMS**

- Collection of software programs
- Manage entire data in database
- Direct communication with hard disk

- **Client Software**

- Communication mediator between user and DBMS
- User send instructions, queries through client software to DBMS.
- DBMS will communicate with Hard Disk and take output from Hard Disk and will return to user through client software

- **Database Server**

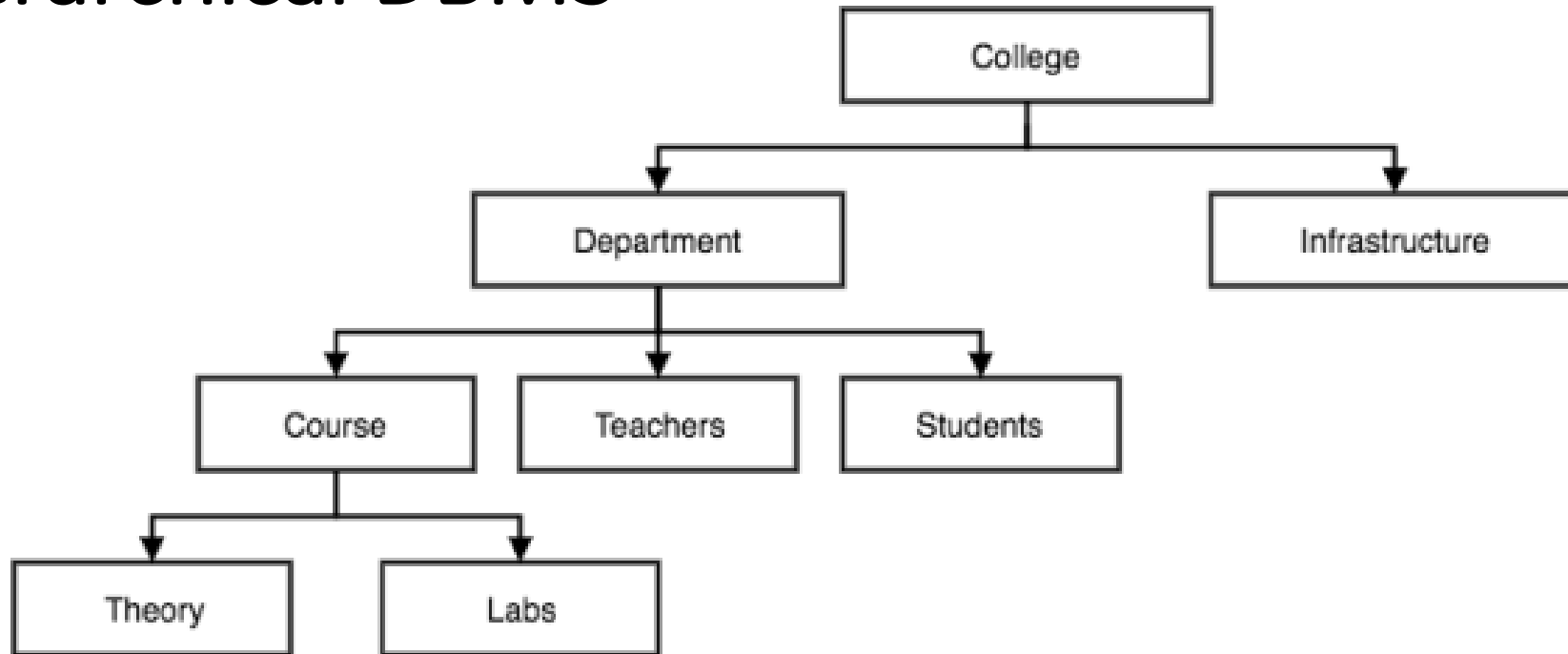
- DBMS + HD (persistent storage)
- This is the back end. You can't see anything from database server.

Types of DBMS

- Based on data, which is saved in backend in hard drive, there are 4 types of database management systems.

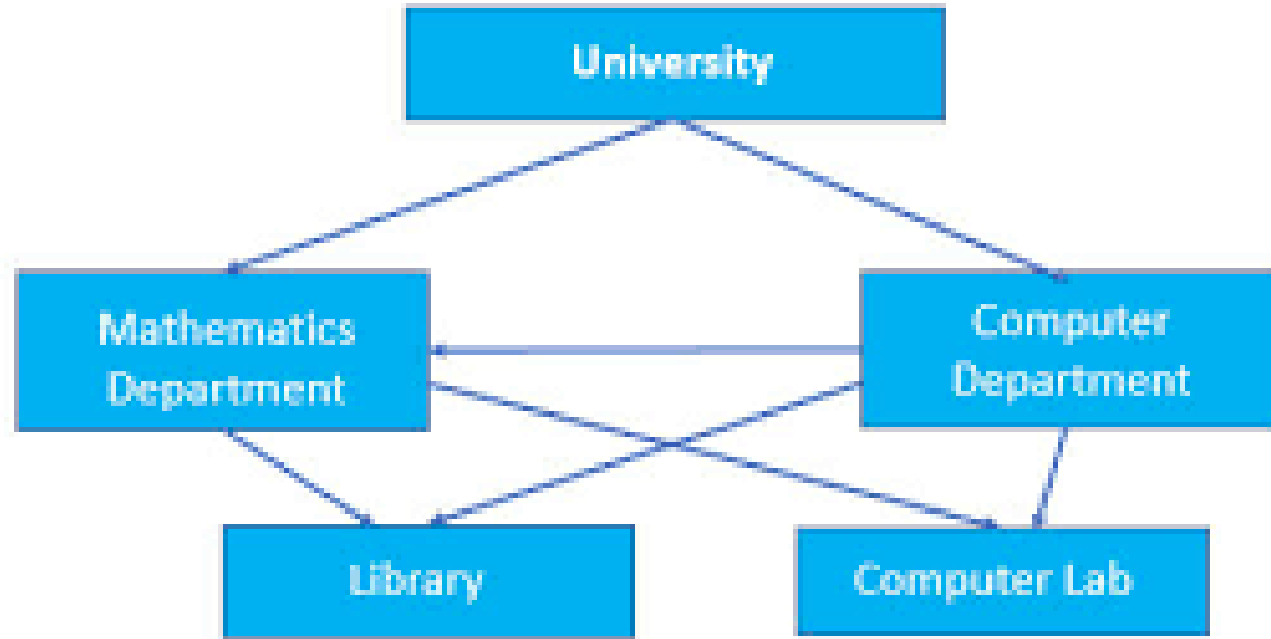


Hierarchical DBMS



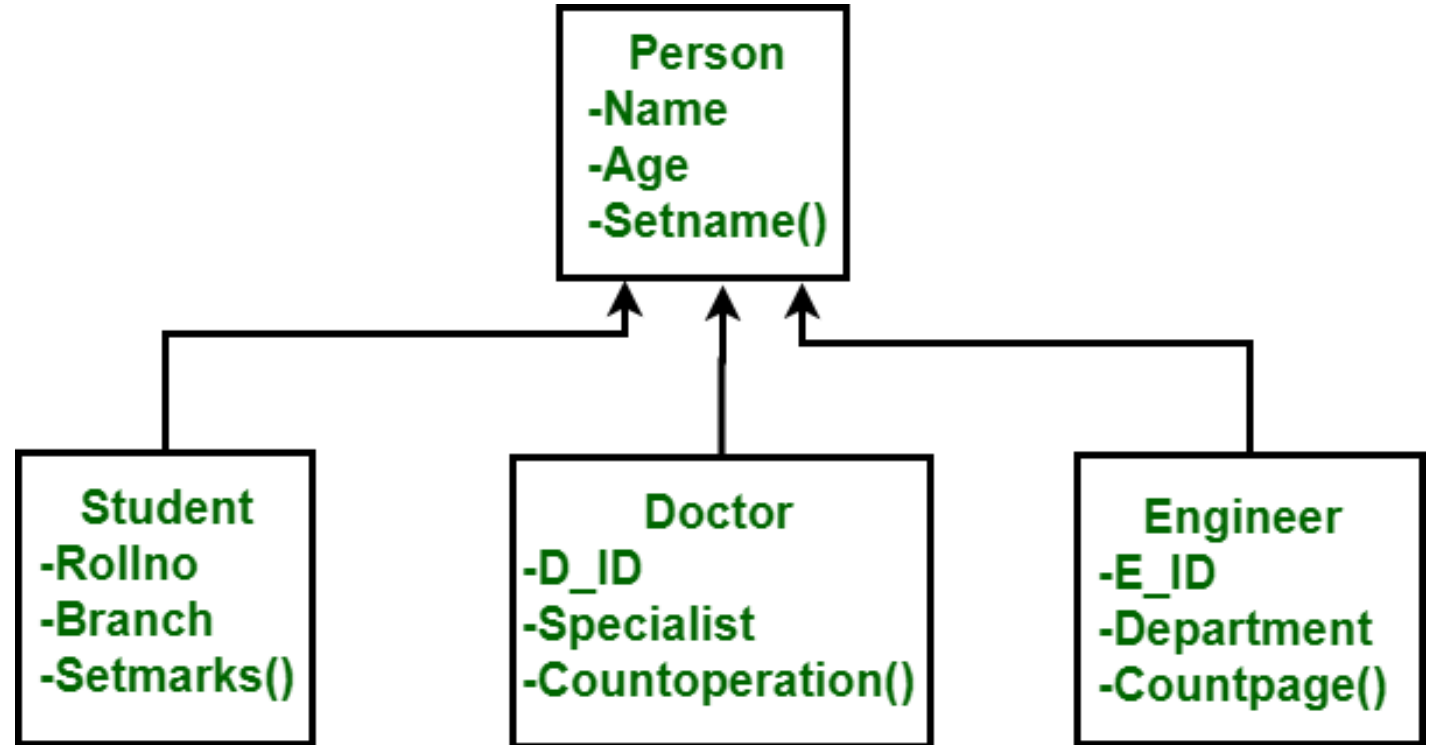
- Instead of just dumping data in database, it is organized in proper (hierarchical) way.
- In this model, data is arranged in a parent-child relationship, where each parent can have multiple children, but each child has only one parent.
- They are efficient for representing certain types of data with clear parent-child relationships, such as organizational structures or file systems.
- They can become complex and difficult to manage when dealing with more interconnected or flexible data relationships.
- Little difficult to store and extract data. Eg. No direct connection between students and course, labs and infrastructure.

Network DBMS



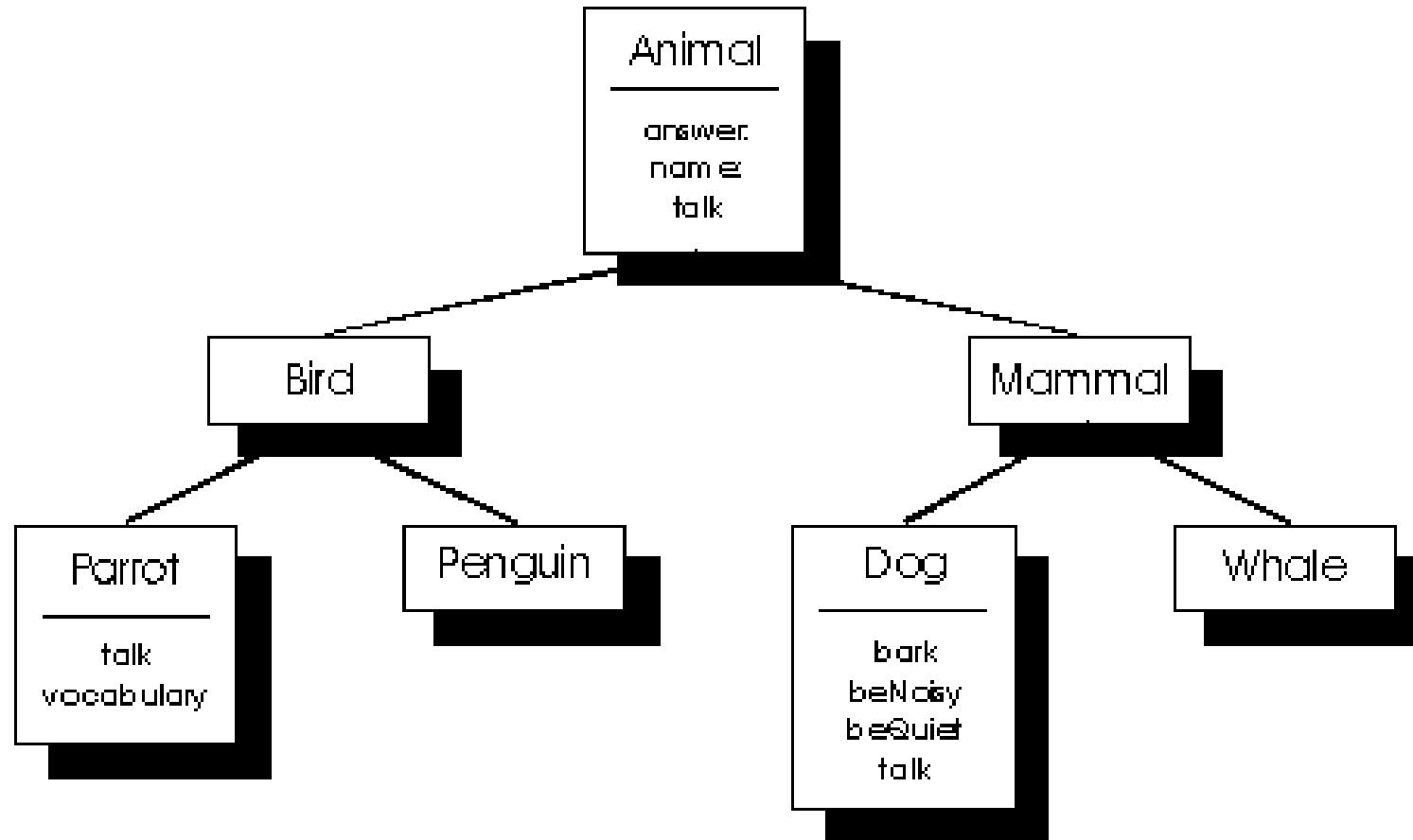
- Network databases allow for more complex relationships between data entities.
- Lot of interconnection between siblings.
- Unlike hierarchical databases, where each node has only one parent, nodes in a network database can have multiple parents, allowing for more flexible data relationships.
- No need to go up and down the hierarchy to find any relation.
- The network model is particularly useful for representing many-to-many relationships, where each entity can be related to multiple other entities and vice versa.
- Network databases are more complex to design and manage compared to hierarchical and relational databases. They require specialized knowledge to navigate and query efficiently.

Object Oriented DBMS



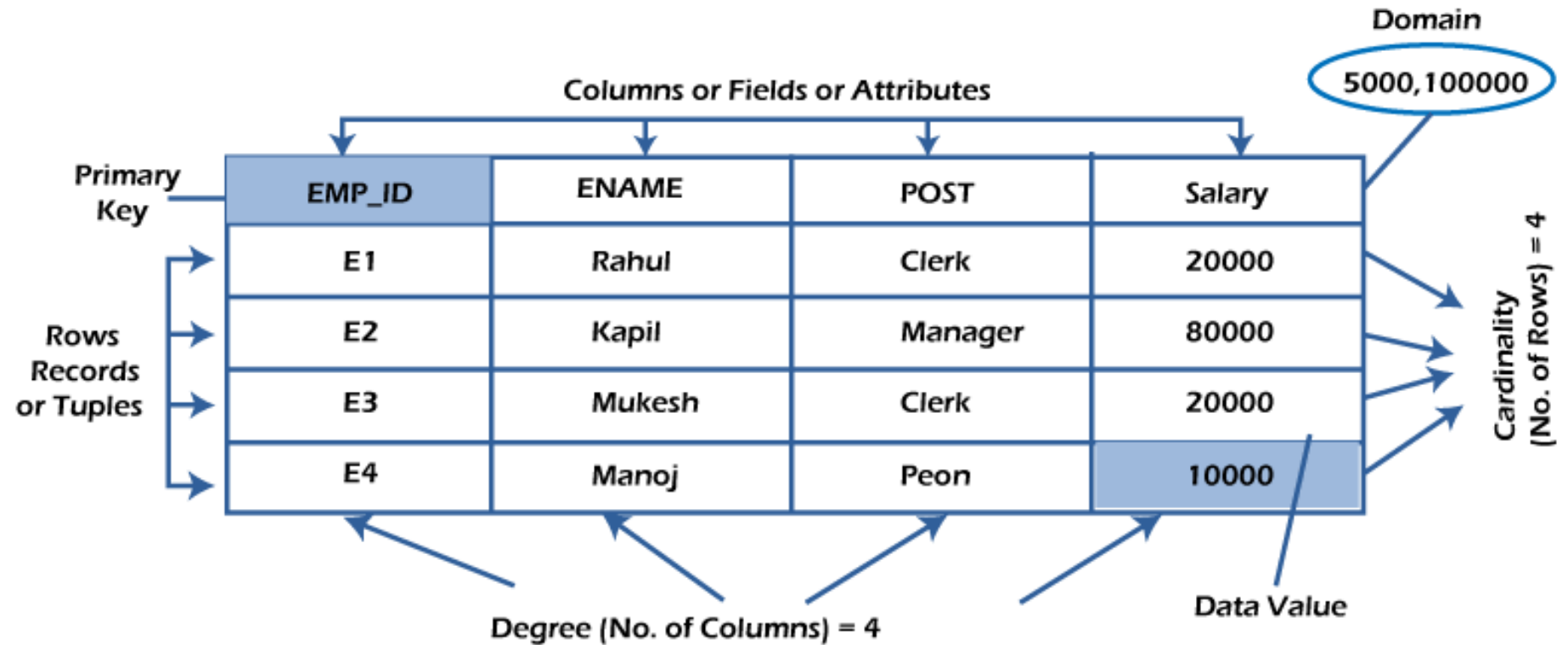
- In an OODBMS, data is represented as objects, which can encapsulate both data and behaviors (methods or functions).
- Base class has all common properties.
- Derived classes inherit properties of base class. It will have its own properties. It is called 'Is a' relationship.
- They can be more complex to design and manage than relational databases, and there may be performance issues with certain types of queries.

Object Oriented DBMS - Inheritance



- To store details of dog(1,Tommy,Male,Lab,4 yrs) + Properties of Mammal + Properties of Animal.
- When we retrieve Tommy we get entire set of data.
- This way of storing and retrieving data is called Object Oriented DBMS.

Relational DBMS



- You can define relation between different tables so it is called as relational. Eg. Student, Course, Faculty

- Companies which implement RDBMS are:
 - Oracle
 - Access
 - SQL Server
 - MySQL
 - Teradata
 - PostgreSQL
- Some companies are good in security or backup or recovery, etc.
- As per our requirement we can choose a product.
- To communicate with Oracle server we need to know the language that Oracle speaks. That language is Structured Query Language (SQL).
- SQL is not a full fledged language like Python but is a set of commands.

- Difference between SQL and MySQL

SQL	VERSUS	MYSQL
SQL		MYSQL
Database language for storing, manipulating and retrieving data in a relational database		Open source Relational Database Management System (RDBMS) that allows managing relational databases
A database language		A software
Helps to manage the data in the relational databases		Helps to manage relational databases using SQL
Does not change as it is a language		Updates frequently as it is a software
		Visit www.PEDIAA.com

- In short MySQL is a product (Server) and SQL is a language.
- If you learn SQL you can work with Oracle, Access, Sql server, MySQL, Teradata, PostgreSQL.
- Unstructured Databases (NoSQL Databases)
 1. Key Stores – Similar to Dictionary in Python. {Key:Value} pair. Eg. Redis
 2. Document Storage – Similar to Key – Value but values can themselves be key – value pairs. Eg. MongoDB
 3. Graph Based Storage - A column-oriented database is a non-relational database that stores the data in columns instead of rows. Eg. Neo4j
 4. Wide Column Storage - Graph-based databases focus on the relationship between the elements. It stores the data in the form of nodes in the database. The connections between the nodes are called links or relationships. Eg. Cassandra, HBase

DBMS

Unstructured DBMS

Key Store - eg. Redis
Document Storage - MongoDB
Graph Based Storage - Neo4j
Wide Column Storage - Cassandra,
HBase

All are NoSQL Databases

Structured DBMS


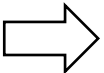
Hierarchical
Network
Object Oriented
Relational

Products

Oracle
Access
SQL Server
Teradata
PostgreSQL
MySQL

Interact with
SQL

Installation of MySQL Server

- In Google type Mysql, select link www.mysql.com
- Click on **Downloads** tab: Scroll down and select: [MySQL Community \(GPL\) Downloads »](#)
- Left side select: [MySQL Installer for Windows](#)
- Select 1st **Download** button in front of Windows (x86, 32-bit), MSI Installer
- Click on: “**No Thanks, just start my download**”. Downloading will start.
- Click on downloaded file to start installation. Allow? **Yes**
- Choose a setup type: Select: **Custom**. Click on **Next**.
- Expand ‘**MySQL Servers**’ by clicking on ‘+’ sign.
- Expand ‘**Server 8.0**’: Choose 1st option 8.0.33
- Click on green arrow 
- To Collapse: click on ‘-’ sign .
- Expand ‘**Applications**’: ‘+’
- Expand ‘**MySQL Workbench**’: ‘+’
- Expand ‘**Workbench 8.0**’: ‘+’: Choose 1st option 8.0.33
- Click on  Then click on ‘**Next**’.
- Click on ‘**Execute**’. **I Agree. Install.** Click on **Next. Execute. Next. Next**

- **Authentication Method**
- Choose 2nd option: '**Use Legacy Auth-Method**'. Click on **Next**.
- Give password. Keep it very simple.
- Click **Next** for 3 times.
- Click **Execute**.
- Click **Finish. Next. Finish.**
- Workbench will open: "**Welcome to MySQL Workbench**".
- Click on '**Local Instance MySQL80**'. Enter Password.
- Top right corner 3 boxes are there. Click on **Last Box**.
- Go to Bottom Left. " is there**No Objects Selected**. Click on it.
- In the middle screen '1' is written.
- Type command: "show databases;"
- Press 'ctrl + enter' to execute the command.
- This is the client we have installed.

Installation of MySQL Server

- Install Client – MySQL Workbench
- Once it is installed execute following query:
- show databases;
- Outputs automatically created databases as follows
 - information_schema – metadata i.e. data about data is stored. eg. # databases, # tables, datatypes, # rows, # columns
 - performance_schema – memory related information, path
 - mysql and sys – configuration related parameters.
- These are all system databases. We will not touch it.
- Create your own databases and work in it.

Thank You!