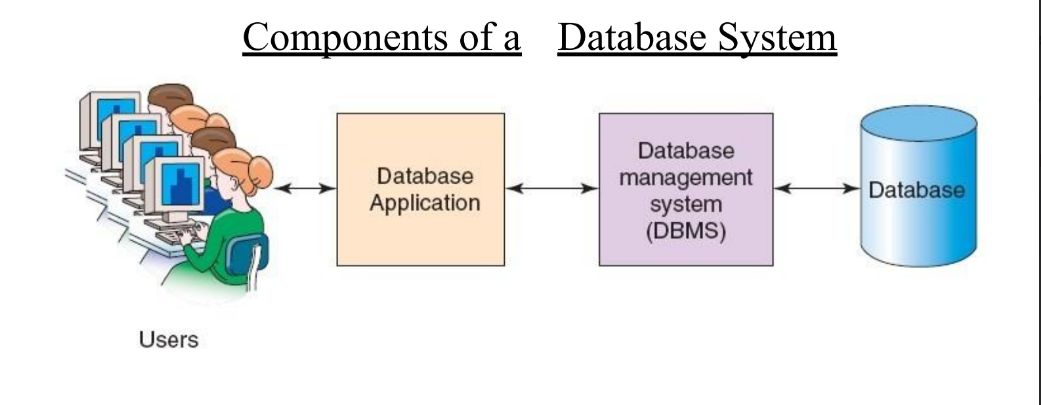
**Day 2**

**Name : Podutur Lahari - DE126**

**Date : 05-11-2024**

**Mysql database**

**1. Introduction to RDBMS**:

* A Database Management System (DBMS) defines, creates, and maintains a database.
* RDBMS (Relational Database Management System) organizes data in tables, fields, and records.

**2. Types of Databases**:

* Examples include operational, distributed, and end-user databases.
* Common RDBMS software includes Oracle, IBM DB2, Microsoft SQL Server, and MySQL.

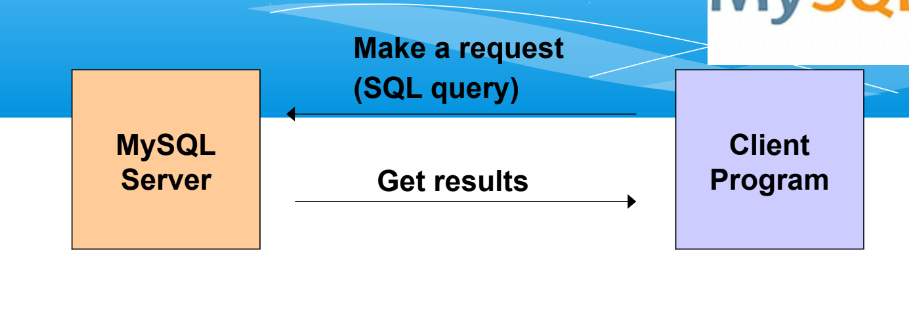
**3. MySQL Overview**:

* MySQL is a popular open-source database known for speed, reliability, and multi-user access.
* Often used in the LAMP stack (Linux, Apache, MySQL, PHP/Perl/Python) for web applications.
* Prominent users include Facebook, Google, Wikipedia, and YouTube.

4. **MySQL Key Features**:

* Written in C/C++ with a yacc-based SQL parser.
* Portable and lightweight, using default port 3306.
* Utilizes different storage engines, with InnoDB as the default for versions after 5.5.

**5. Storage Engines in MySQL**:

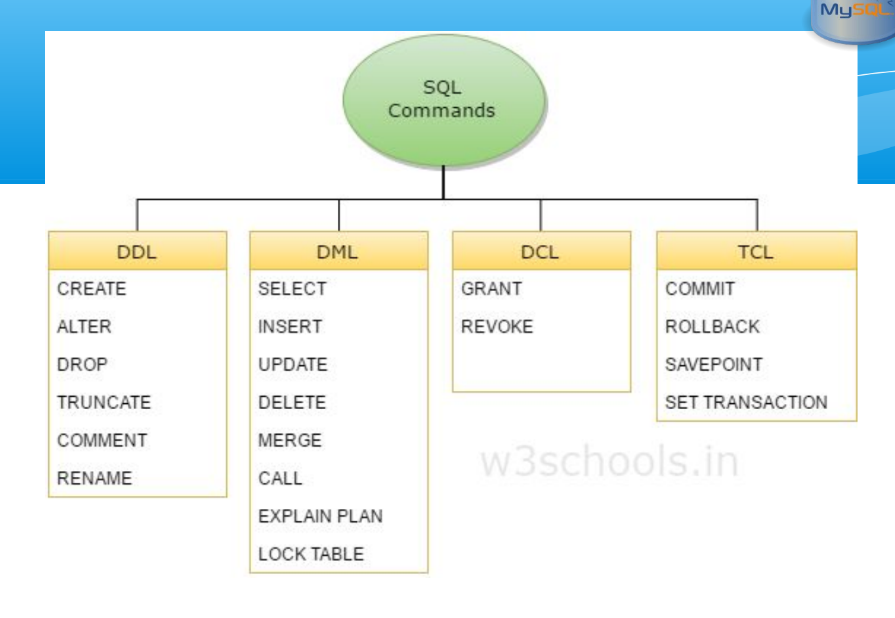
* InnoDB: ACID-compliant, supporting transactions with commit, rollback, and crash recovery.
* Other engines include MyISAM, Memory, Merge, Archive, and CSV.

**6. SQL Basics in MySQL**:

* SQL components:
  + **DDL (Data Definition Language)**: Defines database structure.
  + **DML (Data Manipulation Language)**: Retrieves and updates data.
  + **DCL (Data Control Language)**: Manages access permissions.

**7. MySQL Data Types**:

* **Numeric types**: INT, FLOAT, DECIMAL.
* **Date/Time types**: DATE, DATETIME, TIMESTAMP.
* **String types**: CHAR, VARCHAR, BLOB, ENUM.



**8. Common SQL Commands**:

* **CREATE**: Defines tables and specifies primary keys.
* **DROP**: Deletes tables or databases.
* **INSERT**: Adds rows to a table.
* **SELECT**: Retrieves data from a table, with options to filter and order.
* **UPDATE**: Modifies existing data in a table.

**9. MySQL Client-Server Model**:

* MySQL connectors allow connectivity from programming languages like C, PHP, and Java.
* **Connector/J**: JDBC driver for connecting Java applications to MySQL.

**10. MySQL Cloud SQL**:

* + Google Cloud SQL provides managed MySQL databases with encrypted data, network firewalls, and compatibility with applications like WordPress and e-commerce platforms.

**Data cleaning and Manipulation**

**1. Data Cleaning Basics:**

* Identifying and correcting/removing errors, inaccuracies, and anomalies in a dataset.
* Common issues include:
  + Missing data: Often due to entry errors or incomplete datasets.
  + Incorrect data: Errors from faulty integration or migration processes.
  + Duplicate data: Caused by human error or system glitches.
  + Inconsistent data: Deviations in formats, naming, or units.
  + Outliers: Extreme values affecting analysis.

2. **Impact of Poor Data Quality**:

* Leads to inaccurate insights, misinformed decisions, reduced trust, inefficient resource allocation, and increased costs.
* Investing in data quality processes improves reliability.

3. **SQL Data Cleaning Techniques**:

* SQL’s **declarative approach** makes it ideal for data retrieval and cleaning operations.
* Key SQL syntax for cleaning:
  + **SELECT**: Retrieves data.
  + **WHERE**: Filters data.
  + **UPDATE**: Modifies existing data.
  + **DELETE**: Removes data.
  + **DISTINCT**: Retrieves unique values.
  + **String functions**: Includes TRIM, UPPER, LOWER, and REPLACE for textual data cleaning.
  + **Aggregate functions**: Like COUNT, SUM, AVG, used for identifying outliers or calculating ranges.

4. **SQL Cleaning Techniques**:

* **Removing Duplicates**: Use DISTINCT or group by columns.
* **Handling Missing Values**: Remove rows with nulls or impute with default values.
* **Correcting Inconsistencies**: Use string functions like TRIM, UPPER, and REPLACE.
* **Data Normalization**: Standardize formats using functions (e.g., TO\_DATE for date conversions).
* **Handling Outliers**: Identify through statistics; adjust as needed.
* **Data Integrity**: Enforce primary/foreign key constraints to maintain relationships and prevent invalid data.

5. **Steps in SQL Data Cleaning Process**:

1. **Profiling and Assessment**: Analyze structure and quality, calculate summary statistics.
2. **Data Validation**: Filter erroneous records using specific conditions.
3. **Fixing Missing Data**: Identify and address null values through deletion or imputation.
4. **Standardization and Transformation**: Ensure consistent formats and values.
5. **Removing Duplicates**: Use DISTINCT or group by for unique records.
6. **Correcting Errors**: Use SQL functions to fix inaccuracies and standardize text.
7. **Handling Outliers**: Adjust or remove outliers based on statistical analysis.
8. **Data Integrity Checks**: Use constraints to maintain relationships and enforce data consistency.