MySQL Basics

Great Learning Academy

Agenda

- Database
- Database Management Systems
- Relational Database Management Systems
- Key concepts in RDBMS
- Normalization techniques
- Implementing RDBMS using MySQL

What is Database?

- A database is a container where data can be collected systematically and managing, manipulating of this data is easy.

Example of Database

● Suppose an online telephone directory uses a database to store their data like: Name, address, phone number, other contact details.

Example of Database

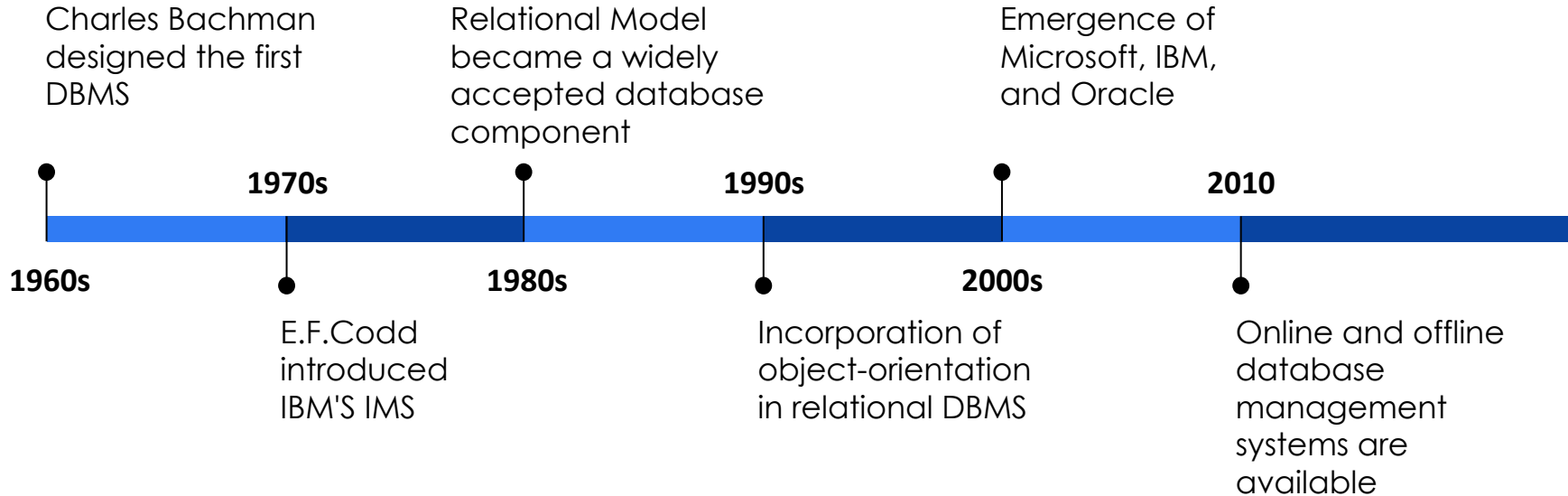- An online library who has millions of books. In order to maintain their data, they use a database.

What is a DBMS?

DBMS also know as Database Management System is:
- a software
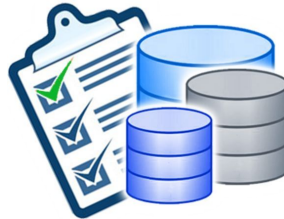- used to store, retrieve, define and manage your data.

History of DBMS

Charles Bachman
designed the first
DBMS

Relational Model
became a widely
accepted database
component

Emergence of
Microsoft, IBM,
and Oracle

1970s   **1990s**   **2010**

1960s   **1980s**   **2000s**

E.F.Codd
introduced
IBM'S IMS

Incorporation of
object-orientation
in relational DBMS

Online and offline
database
management
systems are
available

Commonly used DBMS software

- MSQL
- SQLite
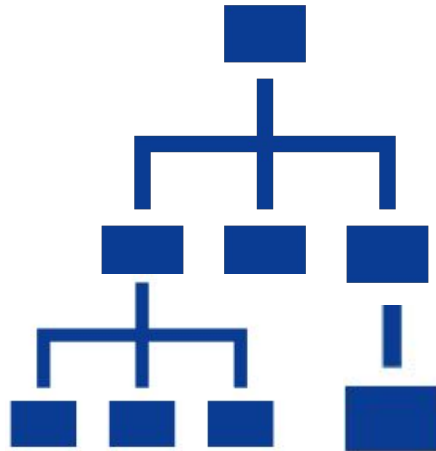- PostgreSQL
- MariaDB
- Oracle

Types of DBMS

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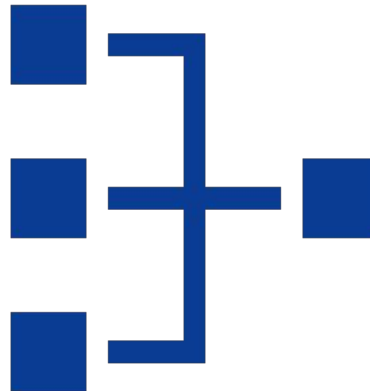- Object Oriented DBMS

Types of DBMS

Hierarchical DBMS
- Tree structure
- 1:N relation
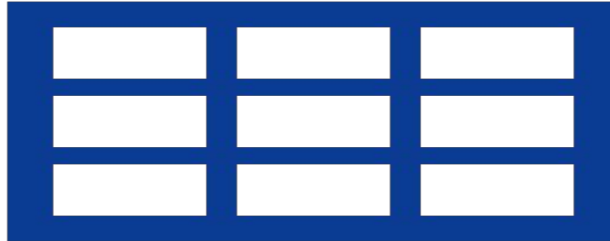- Egs. IMS, Windows registry

Types of DBMS

Network DBMS
- Flexible
- Graph like organization
- 1:1, 1:M, M:N association
- Egs:- Integrated data store,
       Raima database manager

Types of DBMS

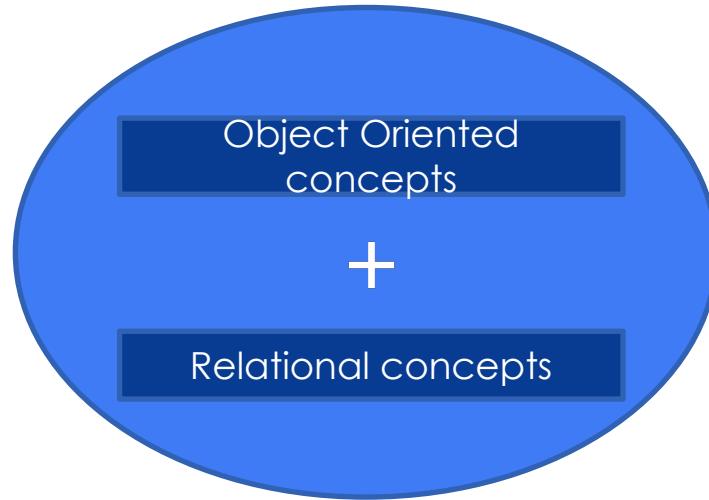**Relational DBMS**

- Widely used
- 2D tables
- Egs: MySQL, Oracle, Microsoft SQL Server

Types of DBMS

Object Oriented DBMS
- Objects
- Combination of relational and OOPs concepts
- Egs: Object DB software

Advantages of using a DBMS

- Efficient data management
- Redundancy reduction
- Data Integrity and security
- Improves data Sharing
- Reduce data processing time
- Improves privacy

DBMS VS File System

File System	DBMS
File system is basically a collection of data and any management with the file system has to write the procedures.	DBMS is a collection of data but user is not required to write the procedures to manage the database.
File systems are not efficient for storing and retrieving of data	DBMS is efficient to use as there are large varieties of techniques to store and retrieve the data.
File system doesn't has crash recovery mechanism. Example: While we are entering some data into the file if System crashes then content of the file is lost	DBMS has a crash recovery option, it protects user from the effects of system failures.
Protecting a file using file system is very difficult.	DBMS has its own security process to save the data

What is RDBMS?

- Relational Database Management System
- Data connected to each other with the help of integrity constraints.

Properties of RDMS

- ❖ Atomic values
- ❖ Rows are entities
- ❖ Columns are properties/attribute
- ❖ Column names are unique

RDBMS vs DBMS

DBMS	RDBMS
Data is stored as files	Data stored as tables
No support for normalization	Constructs for normalization
Minimal security during data manipulation	Integrity constraints for ACID properties
No defined relations between data structures	Defined relation between tables
Small systems	Large systems
Single user support	Multiuser support
Egs, Windows Registry, XML, etc	MySQL, SQL Server etc

Keys in RDBMS- What is a key?

- Columns/set of columns
- identification of records
- Important for relation establishment

Types of keys in RDBMS

Primary key	Super key	Candidate key
Alternate key	Composite Key	Foreign key

Types of keys in RDBMS

Primary key
- Uniquely identify tuples/rows in a table
- No null values
- Unique values
- Single or multiple columns/attributes

Student_ID	Name	Age
S001	Harry	23
S002	Tom	29
S003	James	59

Types of keys in RDBMS

Super key

- One or more columns to uniquely identify tuples
- Superset of candidate key
- Egs, {Student_ID} , {Gr_No}

  {Student_ID, Gr_No}

  {Student_ID, Name}

  {GR_No, Name}

  {Student_ID, Name}

  {Student_ID, GR_No,Name}

Student_ID	GR_No	Name	Age
S001	42005	Harry	23
S002	45678	Tom	29
S003	98456	James	59

Types of keys in RDBMS

Candidate key
- Set of one or more columns that uniquely identify rows in a table
- As strong as primary key
- Egs:- {Gr_No}
        {Student_ID}

Student_ID	GR_No	Name	Age
S001	42005	Harry	23
S002	45678	Tom	29
S003	98456	James	59

Types of keys in RDBMS

Super key vs Candidate key

Super keys	Candidate keys
Superset of candidate key	Subset of super key
Numerically more	Numerically less
Criteria for choosing candidate key	Criteria for choosing primary key

Types of keys in RDBMS

Alternate key

- All others beside the primary key from candidate keys
- Student_ID $\longrightarrow$ **Primary key**
- Passport number
- License number ⎤ **Alternate key**

Student_ID	Name	Passport Number	License number
S001	Harry	CDF56432	8745463
S002	Tom	DF43215	8945297
S003	James	DC55469	9812345

Types of keys in RDBMS

Composite key
- Two or more attributes which together can uniquely identify rows
- No Primary key
- Combination of keys used

First_Name	Last_Name	Class	Subject
Harry	White	3A	Math
Tom	White	3A	Math
Harry	West	3B	Math

Types of keys in RDBMS

Foreign key

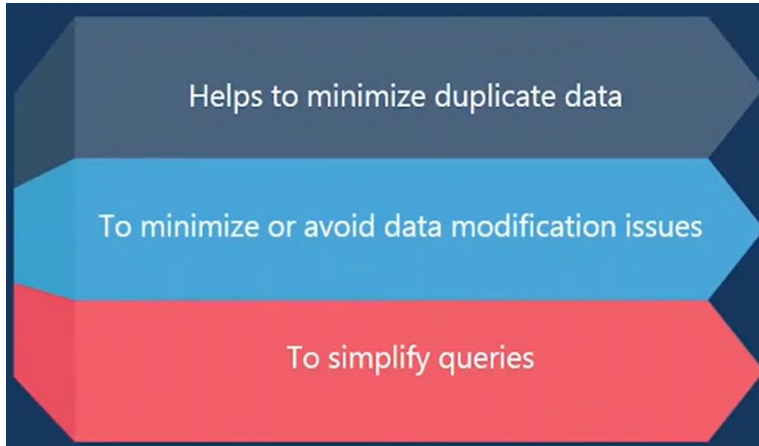- Column in a table referring to the primary key of another table

Student_ID	Name	Passport Number	License number
S001	Harry	CDF56432	87
S002	Tom	DF43215	89
S003	James	DC55469	98

Foreign Key

Student_ID\	Subject
S001	Math
S002	Math
S003	Math

What is Normalization?

- Organizing data
- Avoid redundancy and duplication

Helps to minimize duplicate data

To minimize or avoid data modification issues

To simplify queries

Types of Normalization

First Normal Form

Second Normal Form

Third Normal Form

Boyce-Codd Normal Form

Types of Normalization

First Normal Form

Second Normal Form

Third Normal Form

Boyce-Codd Normal Form

- Atomic:- Single valued
- Unique name for columns
- No importance on order of data storage

Types of Normalization: 1NF

Name	Age	Course
Harry	21	C001, C003
Tom	20	C002
James	19	C003

After using 1NF

Name	Age	Course A	Course B
Harry	21	C001	C003
Tom	20	C002	-
James	19	C003	-

Types of Normalization

First Normal Form

Second Normal Form

Third Normal Form

Boyce-Codd Normal Form

- 1NF
- All columns must be full-functionally dependent on the primary key.

Types of Normalization: 2NF

Student ID	First Name	Project ID	Project Name	Age
S001	Jane	P03	GeoLocator API	20
S002	John	P04	HILaugh	19
S003	John	P05	YFiBlocker	20

After using 2NF

Student ID	First Name	Age
S01	Jane	20
S02	John	19
S03	John	20

Project ID	Project Name
P03	GeoLocator API
P04	HILaugh
P05	YFiBlocker

Types of Normalization

First Normal Form

Second Normal Form

Third Normal Form

Boyce-Codd Normal Form

- 2NF
- No transitive dependencies must exist.

Types of Normalization: 3NF

Student ID	First Name	Course	Fees	Age
S01	Jane	C001	35000	20
S02	John	C002	30000	19
S03	Harry	C002	30000	20

After using 3NF

Student ID	First Name	Age
S01	Jane	20
S02	John	19
S03	Harry	20

Course ID	Course Name	Fees
C01	IT	35000
C02	CS	30000

Types of Normalization

First Normal Form

Second Normal Form

Third Normal Form

Boyce-Codd Normal Form

- Strict 3NF
- 3NF
- All columns fully functionally dependent on the super key.

Types of Normalization: Boyce Codd NF

Student ID	First Name	Course	Professor	Age
S001	Jane	C001	Prof. Chadwick	20
S002	John	C002	Prof. Jacob	21
S003	Harry	C003	Prof June	20

After using BCNF

Student ID	First Name	Age
S01	Jane	20
S02	John	19
S03	Harry	20

Professor ID	Professor Name	Course
P01	Prof. Chadwick	C001
P02	Prof. Jacob	C002
P03	Prof June	C003
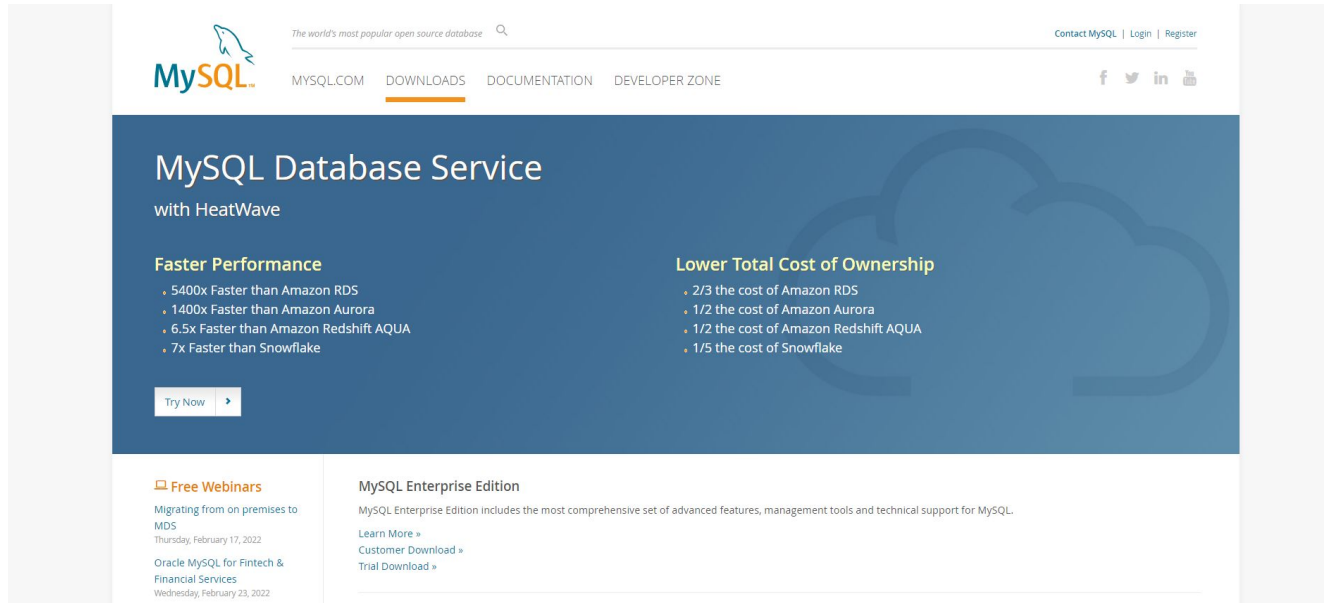
What is SQL? (Implementing RDBMS using MySQL)

- Structured Query Language
- Access and manipulate database
- Standard of:-
- ANSI in 1986
- International organization for standardization in 1987

Installation of MySQL

Site for MySQL Installation:- https://www.mysql.com/downloads/

Data-types in MySQL

- Type of data that an object can hold

Character String	Numeric	Date and Time

Data-types in MySQL

- Character String

Datatype	Description	Length
char(n)	stores n characters	n bytes (where n between 1-8000)
nchar(n)	stores n unicode characters	2n bytes (where n between 1-4000)
varchar(n)	stores approximately n characters	n(String length)+2 bytes (where n between 1-8000)

Data-types in MySQL

- Character String

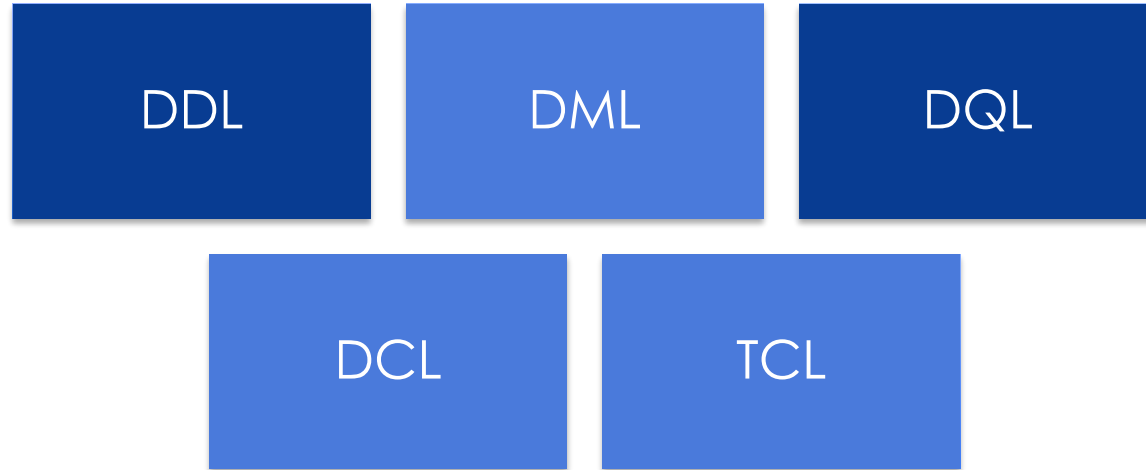| Datatype	Description	Length
char(n)	stores n characters	n bytes (where n between 1-8000)
nchar(n)	stores n unicode characters	2n bytes (where n between 1-4000)
varchar(n)	stores approximately n characters	n(String length)+2 bytes (where n between 1-8000)

Types of Commands in SQL

DDL

DML

DQL

DCL

TCL

Types of Commands in SQL

DDL
- Data Definition Language
- Defines schema of database
- Describing, creating and modifying structure of database

Types of Commands in SQL

DDL Commands
- CREATE - create a new object
- DROP - remove object from database
- ALTER - modify structure of existing object
- TRUNCATE - remove all content from and deallocate memory of a table
- COMMENT - add comment to data dictionary
- RENAME - Rename existing object

Types of Commands in SQL

DML
- Data Manipulation language
- Handles data manipulation

Types of Commands in SQL

DML commands
- Insert- Insert data into a table
- Update- modify data in a table
- Delete:- delete data from a table

Types of Commands in SQL

DQL
- Data Query Language
- Helps query data within a schema object
- Select - command to retrieve data stored in table

Types of Commands

DCL
- Data Control Language
- Rights and permissions on database structures
- GRANT - provide user privileges to database
- REVOKE - withdraw user's access provided by GRANT

Types of Commands in SQL

TCL
- Transaction Control Language
- COMMIT - commits a transaction
- ROLLBACK - undo's a transaction on any error
- SAVEPOINT - to create specific save pointers for systematic rollback and commits
- SET TRANSACTION - specify transactional characteristics

Filters in SQL

- Filter out specific data for viewing
- Where clause
- Syntax:-

Select <column_names> from <table_name> where <condition>;

Operators in SQL

- Combined with where clause to filter data
- AND - returns true if and only if all the conditions are true
- OR - returns true if any one of the conditions is true
- NOT - returns true if the condition is false

Operators in SQL

- Combined with where clause to filter data
- IN - Multiple OR conditions in where clause
- Between - select values between a given range

Pattern Matching in SQL

- "Like" clause
- Used with the "where" clause for pattern matching.

Null values in SQL

- Allows you to have no relevant value without keeping any cell empty.

Summary

- Database
  - What is Database?
  - Example of Database
- DBMS
  - What is a DBMS?
  - History of DBMS;
  - Commonly used DBMS software
  - Types of DBMS
  - Advantages of using a DBMS
- RDBMS
  - What is RDBMS?
  - RDBMS VS DBMS

Summary

- Keys in RDBMS
    - What is a Key?
    - Types of Keys in RDBMS
- Normalization:-
    - What is Normalization?
    - Types of Normalization
- Implementing RDBMS using MySQL:-
    - What is SQL?
    - Implementation in MySQL
    - Types of Commands,
    - Filters, Operators and Pattern Matching
    - Null values

Thank You