upGrad ENTERPRISE

Working with Database : RDBMS Basics

By
Narasimha Rao T
Microsoft.Net FSD Trainer
Professional Development Trainer
tnrao.trainer@gmail.com

Introduction to Databases

1. What is a Database?

Definition:

A **database** is an organized collection of data, generally stored and accessed electronically from a computer system.

Examples:

- Library catalog
- Bank customer records
- Social media user profiles

Database vs File System

Database vs File System

Aspect	File System	Database (DBMS)
Definition	Stores data in files and folders.	A structured system to store, retrieve, and manage data.
Data Redundancy	High, because data may be duplicated in multiple files.	Low, due to normalization and constraints.
Data Integrity	Difficult to maintain.	Enforced through rules, constraints, and relationships.
Querying	Manual search or scripts.	SQL provides efficient querying.
Backup & Recovery	Manual process.	Built-in tools for automated backup and recovery.

Conclusion: Databases are preferred for structured, large-scale, multi-user applications.

Database Management System (DBMS)

Definition:

A **DBMS** is software that interacts with end users, applications, and the database itself to capture and analyze data.

Popular DBMS:

- MySQL
- PostgreSQL
- Oracle
- Microsoft SQL Server
- MongoDB (NoSQL)

Functions of DBMS:

- Data storage, retrieval, and update
- User access control
- Backup and recovery
- Data integrity management
- Query processing

Types of Databases

Type	Description	Example
Relational	Uses tables (relations)	MySQL, SQL Server, PostgreSQL
NoSQL	Non-tabular (key-value, document, etc.)	MongoDB, Cassandra
Hierarchical	Tree-like structure	IBM IMS
Network	Graph-like structure (more flexible)	TurboIMAGE
Object-Oriented	Stores objects like in OOP	db4o

Relational Database Concepts

Key Terms:

- **Table**: Collection of related data in rows and columns
- **Row (Tuple)**: A single record
- **Column (Attribute)**: A field of the record
- **Primary Key**: Uniquely identifies each record
- **Foreign Key**: Links one table to another

RDBMS Concepts : Details

(a) Tables

- A table is a collection of rows and columns.
- Each row is a **record**; each column is an **attribute**.

Example:

StudentID	Name	Age	Department
101	Vinusha	20	Computer Sci
102	Arjun	21	IT

(b) Keys

Keys uniquely identify rows and establish relationships.

Key Type	Description	Example
Primary Key	Unique identifier for each record.	`StudentID` in Student table
Foreign Key	Refers to the primary key of another table.	`DeptID` in Student table
Composite Key	Key formed using multiple attributes.	`(CourseID, StudentID)`
Unique Key	Ensures uniqueness but allows one `NULL`.	`Email`

(c) Relationships

Relationship Type	Description	Example
1:1	One record in a table relates to one in another.	Person ↔️ Passport
1:M	One record relates to many in another.	Department ↔️ Students
M:N	Many records relate to many in another.	Students ↔️ Courses (via junction table)

ER (Entity-Relationship) Modelling

- **Entity:** Object in the real world (e.g., Student, Course).
- **Attributes:** Properties of an entity (e.g., Name, Age).
- **Relationship:** Association between entities.

Example ER Diagram:

```
[Student]----enrolls----<Course>
   |PK: StudentID            |PK: CourseID
   |Name                     |CourseName
   |Age                      |Credits
```

upGrad ENTERPRISE

Introduction to Microsoft SQL Server

SQL Server & SSMS Walkthrough

- **SQL Server:** Relational Database Management System by Microsoft.
- **SSMS (SQL Server Management Studio):** A GUI tool to connect, manage, and query SQL Server databases.

Basic steps:

1. Open SSMS → Connect to Database Engine.
2. Create a new database.
3. Write SQL queries using the Query Editor.
4. Execute queries and view results in the Results window.

SQL Language Categories

Category	Description	Example Command
DDL	Data Definition Language – defines structure.	`CREATE`, `ALTER`, `DROP`
DML	Data Manipulation Language – modifies data.	`INSERT`, `UPDATE`, `DELETE`
DQL	Data Query Language – retrieves data.	`SELECT`

SQL Schema Creation Process

1. **Create a database:**

```
CREATE DATABASE CollegeDB;
```

2. **Use the database:**

```
USE CollegeDB;
```

3. **Create tables:**

```
CREATE TABLE Students (
    StudentID INT PRIMARY KEY,
    Name VARCHAR(50),
    Age INT,
    DeptID INT
);
```

4. **Establish relationships (foreign keys):**

```
CREATE TABLE Departments (
    DeptID INT PRIMARY KEY,
    DeptName VARCHAR(50)
);

ALTER TABLE Students
ADD CONSTRAINT FK_Dept FOREIGN KEY (DeptID) REFERENCES Departments(DeptID);
```

SQL Commands

(a) CREATE

- Creates a database or table.

```
CREATE DATABASE LibraryDB;

CREATE TABLE Books (
    BookID INT PRIMARY KEY,
    Title VARCHAR(100),
    Author VARCHAR(50),
    Price DECIMAL(8,2)
);
```

(b) ALTER

- Modifies an existing table.

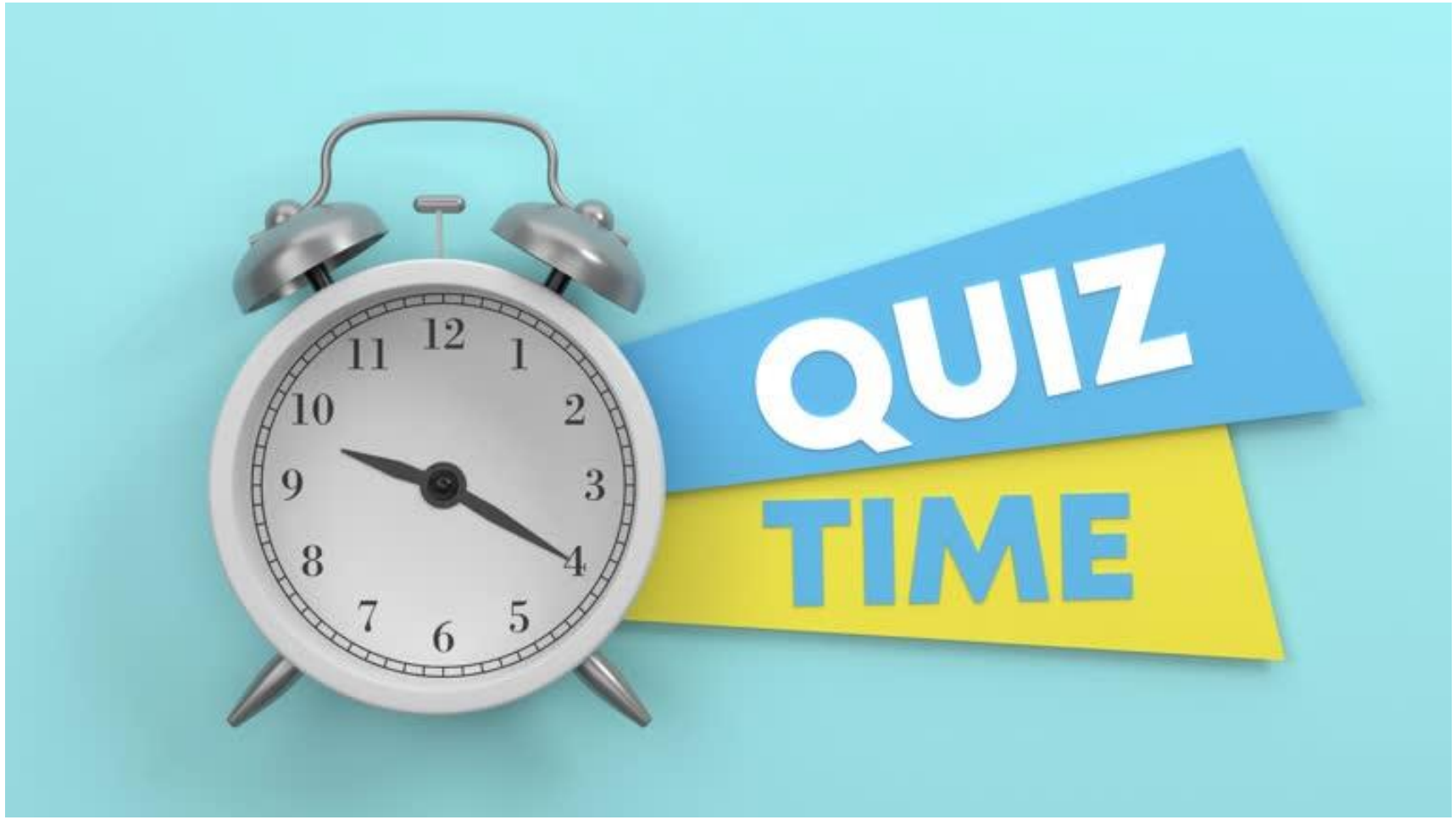
```
ALTER TABLE Books
ADD Publisher VARCHAR(50);

ALTER TABLE Books
DROP COLUMN Publisher;
```

(c) DROP

- Deletes a table or database.

```
DROP TABLE Books;
DROP DATABASE LibraryDB;
```


QUIZ
TIME

Quiz Questions

1. What is the difference between a **database** and a **file system**?
2. Explain **Primary Key** and **Foreign Key** with examples.
3. What are the three types of relationships in ER modeling?
4. Write SQL to:
    - Create a table `Employees` with `EmpID`, `Name`, and `Salary`.
    - Add a column `Department`.
5. What is the difference between **DDL** and **DML**?
6. How do you establish a foreign key relationship between two tables?

upGrad ENTERPRISE

Q & A

Narasimha Rao T
tnrao.trainer@gmail.com