# Database Management System (DBMS)

## Lecture-1

Dharmendra Kumar

August 9, 2020

# Syllabus

**Unit-I:**

Introduction: Overview, Database System vs File System, Database System Concept and Architecture, Data Model Schema and Instances, Data Independence and Database Language and Interfaces, Data Definitions Language, DML, Overall Database Structure. Data Modeling Using the Entity Relationship Model: ER Model Concepts, Notation for ER Diagram, Mapping Constraints, Keys, Concepts of Super Key, Candidate Key, Primary Key, Generalization, Aggregation, Reduction of an ER Diagrams to Tables, Extended ER Model, Relationship of Higher Degree.

# Syllabus

**Unit-II:**

Relational data Model and Language:Relational Data Model Concepts, Integrity Constraints, Entity Integrity, Referential Integrity, Keys Constraints, Domain Constraints, Relational Algebra, Relational Calculus, Tuple and Domain Calculus. Introduction on SQL: Characteristics of SQL, Advantage of SQL. SQl Data Type and Literals. Types of SQL Commands. SQL Operators and Their Procedure. Tables, Views and Indexes. Queries and Sub Queries. Aggregate Functions. Insert, Update and Delete Operations, Joins, Unions, Intersection, Minus, Cursors, Triggers, Procedures in SQL/PL SQL

**Unit-III:**

Data Base Design Normalization:Functional dependencies, normal forms, first, second, 8 third normal forms, BCNF, inclusion dependence, loss less join decompositions, normalization using FD, MVD, and JDs, alternative approaches to database design

# Syllabus

**Unit-IV:**

Transaction Processing Concept: Transaction System, Testing of Serializability, Serializability of Schedules, Conflict View Serializable Schedule, Recoverability, Recovery from Transaction Failures, Log Based Recovery, Checkpoints, Deadlock Handling. Distributed Database: Distributed Data Storage, Concurrency Control, Directory System.

**Unit-V:**

Concurrency Control Techniques:Concurrency Control, Locking Techniques for Concurrency Control, Time Stamping Protocols for Concurrency Control, Validation Based Protocol, Multiple Granularity, Multi Version Schemes, Recovery with Concurrent Transaction, Case Study of Oracle.

## Text books

1. Korth, Silbertz, Sudarshan," Database System Concepts", McGraw Hill
2. Elmasri, Navathe, " Fundamentals of Database Systems", Addision Wesley

# Course Outcome

| | |
|---|---|
| CO 1 | Identify the basic concepts and various data model used in database, design ER modelling concepts and architecture use. |
| CO 2 | Apply relational database theory and be able to describe relational algebra expression, tuple and domain relation expression fro queries and design queries using SQL. |
| CO 3 | Identify and solve the redundancy problem in database tables using normalization. |
| CO 4 | Understand the concepts of transactions, their processing so they will familiar with broad range of database management issues including data integrity, security and recovery. |
| CO 5 | Illustrate and compare the concurrency control protocols and explain recovery techniques. |

# Database

- It is a collection of interrelated data of an enterprise in a particular subject.
- It is a collection of data in an organized manner in a persistent media so that storing and retrieving data will be easier.

**For example,** a university database might contain information about the following:-
**Entities** such as students, faculty, courses, and classrooms.
**Relationships between entities**, such as students' enrollment in courses, faculty teaching courses, and the use of rooms for courses.

# Database Management System (DBMS)

- It is a collection of programs(software), which is used to create, manipulate(insert, delete, retrieve, update) data in database.
- The primary goal of a DBMS is to provide a way to store and retrieve database information that is both convenient and efficient.

# Applications of DBMS

Databases are widely used. Here are some representative applications:

- **Banking:** In banking database, we record the information about customers, accounts, loans and banking transactions.
- **Railway reservation system:** we record the information about trains, reservation, passengers.
- **Airlines:** We record reservations and schedule information, flight information.
- **Universities:** We record student information, course registrations, and grades.
- **Credit card transactions:** We record purchases on credit cards and generation of monthly statements.

# DBMS

**Telecommunication:** For keeping records of calls made, generating monthly bills, maintaining balances on prepaid calling cards, and storing information about the communication networks.

**Sales:** For customer, product, and purchase information.

**Human resources:** For information about employees, salaries, payroll taxes and benefits, and for generation of paychecks.

**Hospital system:** We record information about doctors, patients, services available, rooms details, employees etc.

**Hotel system:** We record information about rooms,customers and employees details.

## Drawbacks of File system

To keep information in such file-processing system, there are a number of major disadvantages:-

**Data redundancy:** Data redundancy refers to the duplication of data, lets say we are managing the data of a college where a student is enrolled for two courses, the same student details in such case will be stored twice, which will take more storage than needed. Data redundancy often leads to higher storage costs and poor access time.

**Data inconsistency:** Data redundancy leads to data inconsistency, lets take the same example that we have taken above, a student is enrolled for two courses and we have student address stored twice, now lets say student requests to change his address, if the address is changed at one place and not on all the records then this can lead to data inconsistency.

**Difficulty in accessing data:** In file system the data is stored in the files. Whenever data has to be retrieved as per the requirements then a new application program has to be written. This is tedious process.

# DBMS

**Data isolation:** Because data are scattered in various files, and files may be in different formats, writing new application programs to retrieve the appropriate data is difficult.

**Integrity problems:** The data values stored in the database must satisfy certain types of consistency constraints.

For example, the balance of a bank account may never fall below a prescribed amount. These constraints are enforced in the system by adding appropriate code in the various application programs. However, when new constraints are added, it is difficult to change the programs to enforce them. The problem is compounded when constraints involve several data items from different files.

**Atomicity problems:** Atomicity of a transaction refers to "All or nothing", which means either all the operations in a transaction executes or none.

For example: Lets say Steve transfers 100$ to Negan's account. This transaction consists multiple operations such as debit 100$ from Steve's account, credit 100$ to Negan's account. Like any other device, a computer system can fail lets say it fails after first operation then in that case Steve's account would have been debited by 100$ but the amount was not credited to Negan's account, in such case the rollback of operation should occur to maintain the atomicity of transaction. It is difficult to achieve atomicity in file processing systems.

**Concurrent-access anomalies:** Concurrent access means multiple users can access database simultaneously. In such an environment, interaction of concurrent updates may result in inconsistent data.

Consider bank account A, containing $500. If two customers withdraw funds (say $50 and $100 respectively) from account A at about the same time, the result of the concurrent executions may leave the account in an incorrect (or inconsistent) state. There is no central control of data in classical file organization. So, the concurrent access of data by many users is difficult to implement.

**Security problems:** Data should be secured from unauthorised access, for example a student in a college should not be able to see the payroll details of the teachers, such kind of security constraints are difficult to apply in file processing systems.

Not every user of the database system should be able to access all the data.