Database Management Systems (DBMS)

RAJAD SHAKYA

Syllabus

1. **Introduction [3 hours]**

    1.1. Concepts and Applications
    1.2. Objective and Evolution
    1.3. Data Abstraction and Data Independence
    1.4. Schema and Instances
    1.5. Concepts of DDL, DML and DCL

Syllabus

2. Data Models [7 hours]

2.1. Logical, Physical and Conceptual
2.2. E-R Model
2.3. Entities and Entities sets
2.4. Relationship and Relationship sets
2.5. Strong and Weak Entity Sets
2.6. Attributes and Keys
2.7. E-R Diagram
2.8. Alternate Data Model (hierarchical, network, graph)

Syllabus

3. Relational Languages and Relational Model [7 hours]

3.1. Introduction to SQL
3.2. Features of SQL
3.3. Queries and Sub-Queries
3.4. Set Operations
3.5. Relations (Joined, Derived)
3.6. Queries under DDL and DML Commands
3.7. Embedded SQL

Syllabus

3. Relational Languages and Relational Model [7 hours]

3.8. Views
3.9. Relational Algebra
3.10. Database Modification
3.11. QBE and domain relational calculus

Syllabus

4. Database Constraints and Normalization [6 hours]

4.1. Integrity Constraints and Domain Constraints
4.2. Assertions and Triggering
4.3. Functional Dependencies (Chase Algorithm)
4.4. Multi-valued and Joined Dependencies
4.5. Different Normal Forms (1st, 2nd, 3rd, BCNF, DKNF)

Syllabus

5. Query Processing and Optimization [4 hours]

5.1. Query Cost Estimation
5.2. Query Operations
5.3. Evaluation of Expressions
5.4. Query Optimization
5.5. Query Decomposition
5.6. Performance Tuning

Syllabus

6. File Structure and Hashing [4 hours]

6.1. Records Organizations
6.2. Disks and Storage
6.3. Remote Backup System
6.4. Hashing Concepts, Static and Dynamic Hashing
6.5. Order Indices
6.6. B+ tree index

Syllabus

7. **Transactions processing and Concurrency Control [6 hours]**

    7.1. ACID properties
    7.2. Concurrent Executions
    7.3. Serializability Concept
    7.4. Lock based Protocols
    7.5. Deadlock handling and Prevention

Syllabus

8. Crash Recovery [4 hours]

8.1. Failure Classification
8.2. Recovery and Atomicity
8.3. Log-based Recovery
8.4. Shadow ept of Spatial Database
8.5. Advanced Recovery Techniques

Syllabus

9. **Advanced database Concepts [4 hours]**

    9.1. Concept of Object-Oriented and Distributed Database Model
    9.2. Properties of Parallel and Distributed Databases
    9.3. Concept of Data warehouse Database

Practicals

1. PostgreSQL & PgAdmin Installation & Setup
2. Intro to SQL & DDL Commands
3. DML Commands in SQL + Select with conditions
4. Grouping, Filtering, Having + DCLs
5. Joins in SQL
6. Python + SQL Connection (Embedded SQL)

Practicals

7. Project Proposal
8. Subqueries & Procedures / Triggers
9. Transactions & More Advanced Topics
10. Project Defense

Introduction

- Data

Introduction

- Data
    - Raw, unprocessed facts, figures, symbols, or characters.
    - It has no inherent meaning on its own.
    - Eg: A string of digits like "10247"
    - Eg: temperature readings (e.g., 22°C, 23°C, 21°C)

Introduction

- Information

Introduction

- Information
  - Data that has been processed and organized to convey meaning
  - It provides context and understanding to raw data.
  - Eg: 10247 - product code
  - Eg: Room temperature at 2 PM: 22°C

Introduction

- Database

Introduction

- Database
  - A structured collection of information, typically stored electronically in a computer system.
  - It allows for efficient storage, retrieval, and management of large datasets.
  - Eg: A library database - information about books, including titles, authors, genres

Introduction

- DBMS (Database Management System)

Introduction

- DBMS (Database Management System)
    - Software that allows users to create, manage, and interact with databases
    - It provides tools for defining the structure of a database, manipulating data.
    - Eg:MySQL or PostgreSQL to create and manage database

Applications

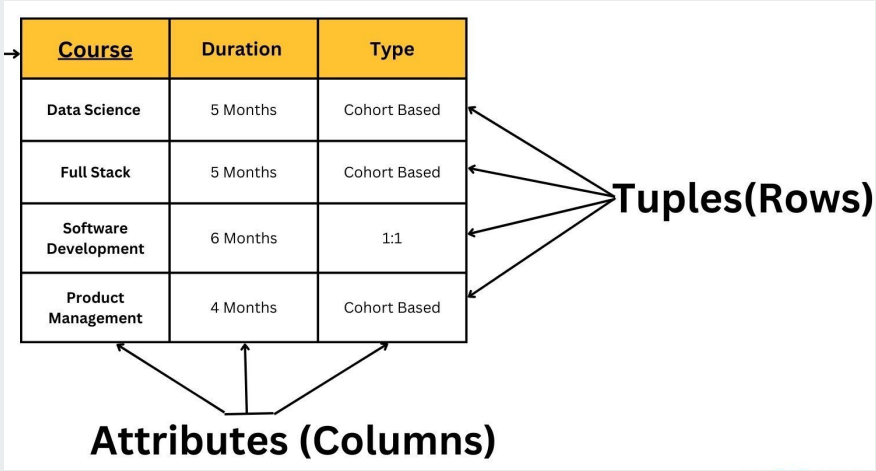
- Business Applications:
  Customer relationship management (CRM), inventory control
- E-commerce:
  Product information, customer data, order processing
- Social Networks:
  User profiles, connections, messages, activity feeds.
- Healthcare:
  Patient records, medical history, appointment scheduling.

Assignment

- How does twitter use database systems to store data ?

Types of Database

- Relational Databases (RDBMS)
  - Data is stored in tables with rows and columns.
  - Each table represents a specific entity, and columns represent attributes of that entity.



Types of Database

- Non-Relational Databases (NoSQL):
    - Data can be stored in various formats like documents (JSON), key-value pairs, graphs, or wide-column stores.

Types of Database

- Non-Relational Databases (NoSQL):
    - Data can be stored in various formats like documents (JSON), key-value pairs, graphs, or wide-column stores.

DBMS vs File System

- Data Redundancy
- Data Inconsistency
- Difficult Data Sharing
- Limited Data Security
- Inefficient Query Processing
- Limited Data Integrity
- No Concurrency Control
- Limited Recovery Options

Lets see a example

Students

id	name	college	location
1	a	ismt	ktm
2	b	herald	lal
3	c	british	bhak
4	d	ismt	ktm
5	e	ismt	ktm

Lets see a example

Students

id	name	college_id(fk)
1	a	1
2	b	2
3	c	3
4	d	1
5	e	1

College

id (pk)	college	location
1	ismt	ktm
2	herald	lal
3	british	bhak

Lets see a example

Students

id	name	college	location
1	a	ismt	ktm
2	b	herald	lal
3	c	british	bhak
4	d	ismt	ktm
5	e	ismt	ktm
6	f	islington	pokh

Lets see a example

Students

id	name	college_id(fk)
1	a	1
2	b	2
3	c	3
4	d	1
5	e	1
6	f	4

College

id (pk)	college	location
1	ismt	ktm
2	herald	lal
3	british	bhak
4	islington	pokh

Try this !!

ID	FirstName	LastName	Class	Department	Department_Code
1	John	Doe	10	Science	101
2	Jane	Smith	11	Arts	102
3	Mark	Johnson	12	Science	101

Try this !!

Departments

Department	Department_Code (PK)
Science	101
Arts	102

Students

ID	FirstName	lastName	Grade	Department_Code (FK)
1	John	Doe	10	101
2	Jane	Smith	11	102
3	Mark	Johnson	12	101

Data Abstraction

Data Abstraction

- hides unnecessary data storage details from users and applications
- exposing only the essential information needed for interaction.
-

Types of Data Abstraction:

- Physical Level (Internal Level):
  - lowest level of abstraction
  - dealing with the physical storage of data on disks
  - details like data structures, access methods, and storage allocation

Types of Data Abstraction:

- Logical Level (Conceptual Level)
  - Defines the logical structure of the data
  - focusing on what data is stored and how it's organized
  - uses data models to represent entities, attributes, and relationships between them.
  - Users and applications interact with the database at this level

Types of Data Abstraction:

- View Level (External Level)
    - Provides different user groups with customized views of the data.
    - Views can be based on tables, joins, and queries
    - simplified and focused representation of the underlying data

Types of Data Abstraction:

- Eg: library database
    - Librarians
    - Patrons

Data Independence

- ability to modify the schema (structure) of a database at one level
- without affecting the schema or functionality at a higher level.
- helps maintain data integrity and reduces the need to rewrite application logic

Data Independence

- ability to modify the schema (structure) of a database at one level
- without affecting the schema or functionality at a higher level.
- helps maintain data integrity and reduces the need to rewrite application logic

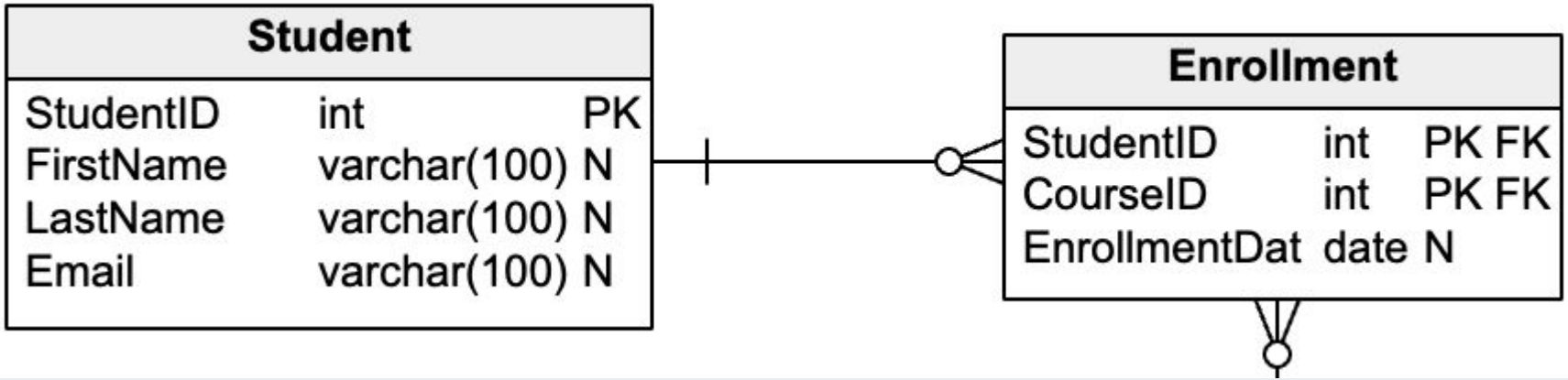
Types of Data Independence

- Physical Data Independence:
    - ability to modify the physical storage details of a database without affecting the logical level
    - Changing the storage device (e.g., from hard drive to SSD)

Types of Data Independence

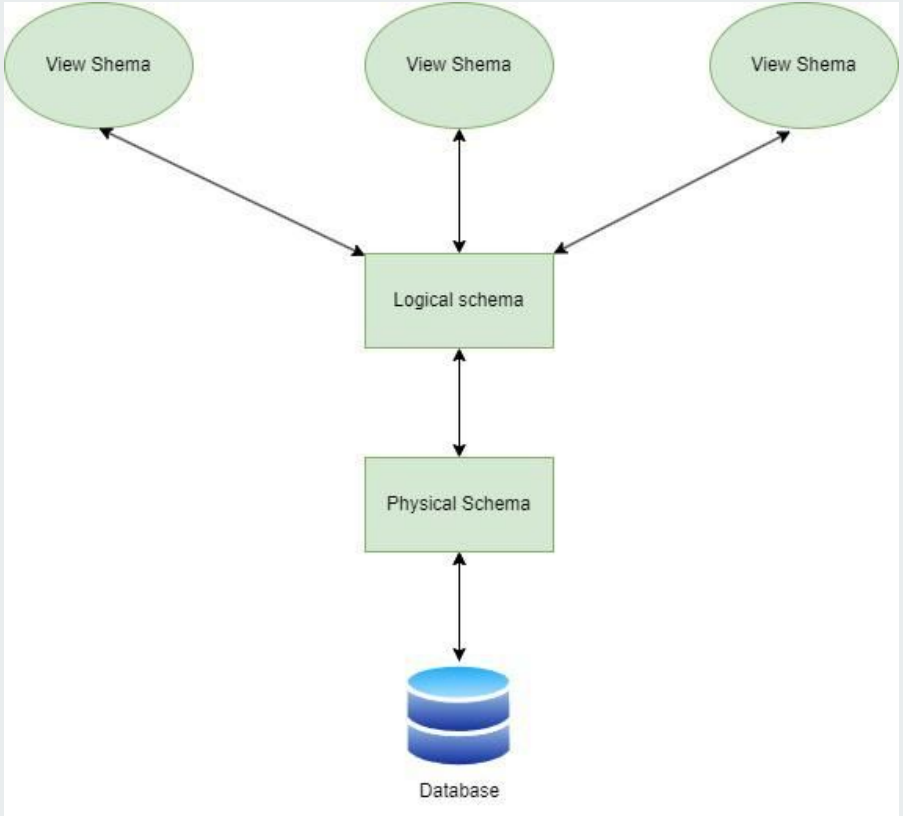
- Logical Data Independence:
    - ability to modify the logical structure of a database without affecting the applications
    - Eg: adding new columns, removing existing ones, changing data types

Schema



- logical structure, acts as a blueprint for your database.
- It includes Tables, Columns, Data Types, Constraints

Types of Schema



Types of Schema

- Physical Schema:
    - deals with the physical storage details of the data,
    - including storage allocation, indexing methods, and access paths.
    - typically hidden from users as it deals with the internal workings of the DBMS.

Types of Schema

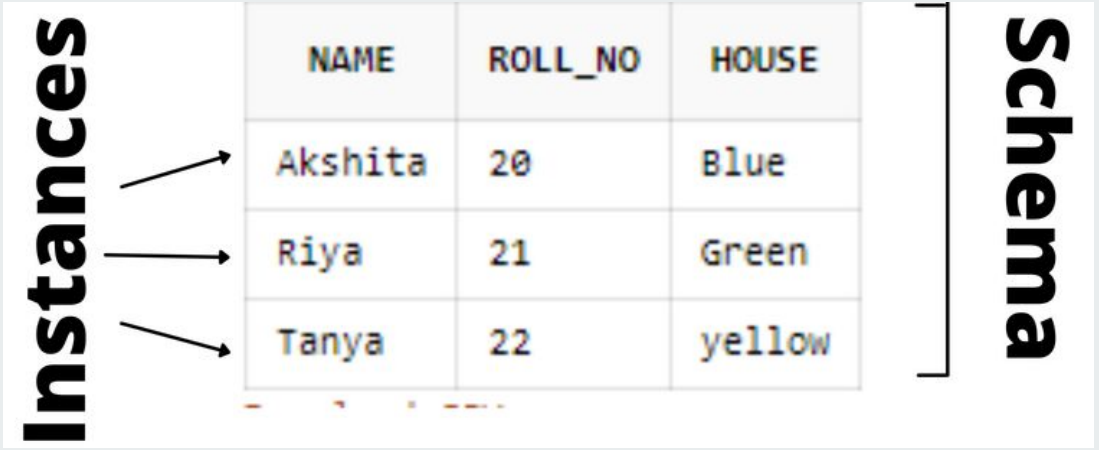
- Logical Schema
    - user's view of the database, focusing on the tables, columns, data types, and relationships.
    - It describes how data is logically organized and accessed.

Types of Schema

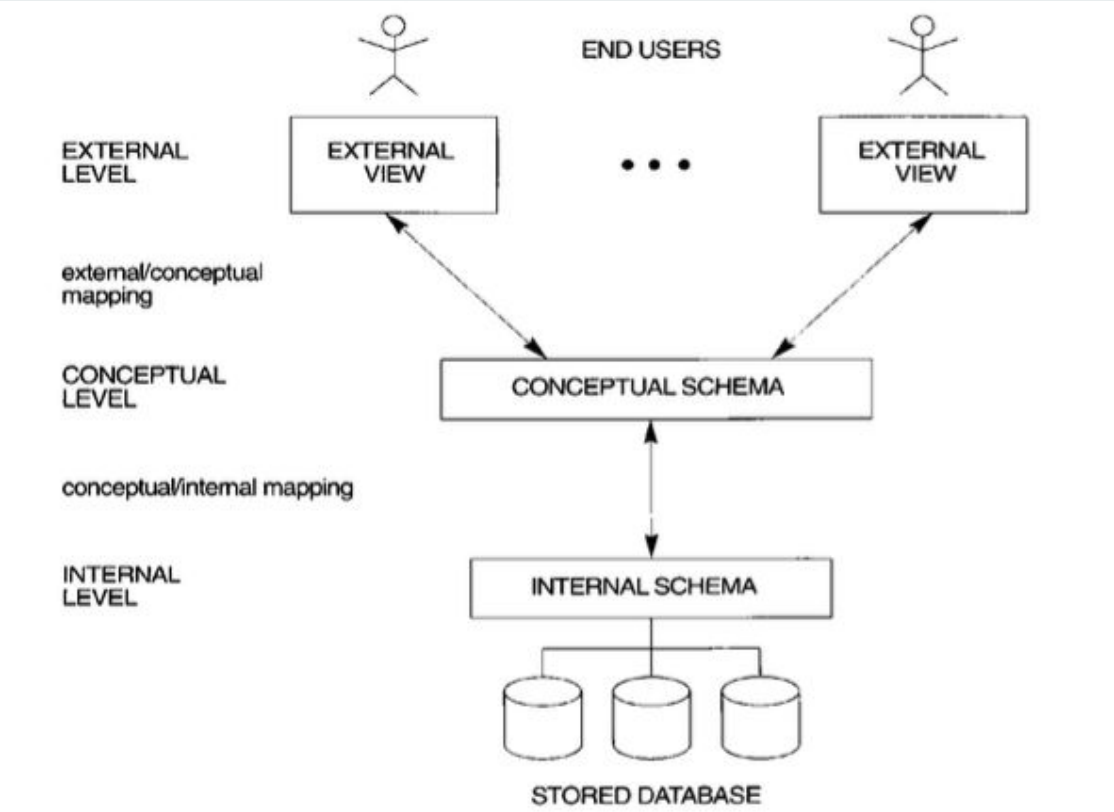
- View Schema
  - virtual table created based on a query against the underlying tables.
  - It allows users to see a specific subset of data
  - present the data in a customized way without modifying the base tables

Instance

- is the actual data stored in the database at a specific point in time.
- It represents the population of the tables defined by the schema.



Three-Schema Architecture



Thank You

RAJAD SHAKYA