Database Management System (DBMS)

Module Name: **Relational Database Management System**

Module Code: **CS 1211**

Instructors: **Jamila L.**



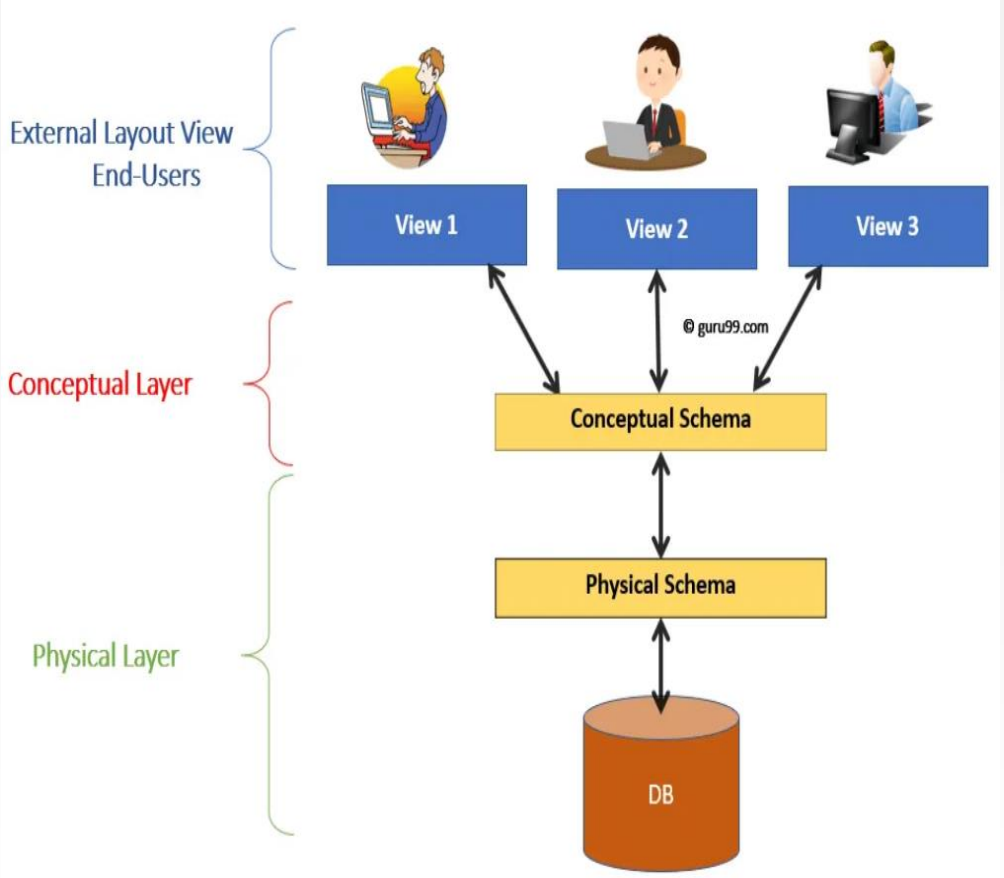
DBMS Architecture

DBMSs do not all confirm to the same architecture.

- The three-level architecture forms the basis of modern database architectures.
- The architecture for DBMSs is divided into three general levels:
  1. External Schema
  2. Conceptual Schema
  3. Internal Schema

DBMS Architecture cont…

1. The external level : concerned with the way individual users see the data
2. The conceptual level : can be regarded as a community user view a formal description of data of interest to the organisation, independent of any storage considerations.
3. The internal level : concerned with the way in which the data is actually stored



RDBMS Models

Data Models

Data Models are fundamental entities to introduce abstraction in a DBMS. Data models define how data is connected to each other and how they are processed and stored inside the system.

- A Data Model in Database Management System (DBMS) is the concept of tools that are developed to summarize the description of the database.
- Data Models provide us with a transparent picture of data which helps us in creating an actual database. It shows us from the design of the data to its proper implementation of data.

Relational Model

The Relational Model organizes data using tables (relations) consisting of rows and columns.

- The relational model represents how data is stored and managed in Relational Databases where data is organized into tables, each known as a relation.
- Each row of a table represents an entity or record and each column represents a particular attribute of that entity.

Types of Relational Models

i. Physical Data Model

ii. Representational Data Model

iii. Conceptual Data Model

Relational Model: Physical Data Model

- The physical Data Model is used to practically implement Relational Data Model. Ultimately, all data in a database is stored physically on a secondary storage device such as discs and tapes.
- This is stored in the form of files, records, and certain other data structures.
- Here, we basically save tables in memory so they can be accessed efficiently.
- In order to come up with a good physical model, we have to work on the relational model in a better way.
- Structured Query Language (SQL) is used to practically implement Relational Algebra for physical data model.

Relational Model: Representational Data Model

This type of data model is used to represent only the logical part of the database and does not represent the physical structure of the database.

- The advantage of using a Representational data model is to provide a foundation to form the base for the Physical model.

Relational Model: Conceptual Data Model

The conceptual data model describes the database at a very high level and is useful to understand the needs or requirements of the database.

- It is this model, that is used in the requirement-gathering process i.e. before the Database Designers start making a particular database.
- One such popular model is the entity/relationship model (ER model).
- The relational model transforms conceptual designs from ER diagrams into implementable structures. These structures are used in relational database systems like Oracle SQL and MySQL.

Relational Model: Entity-Relationship Model(ER Model)

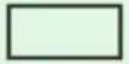
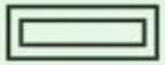
Entity-Relationship Model(ER Model): It is a high-level data model which is used to define the data and the relationships between them. It is basically a conceptual design of any database which is easy to design the view of data. Components of ER Model:

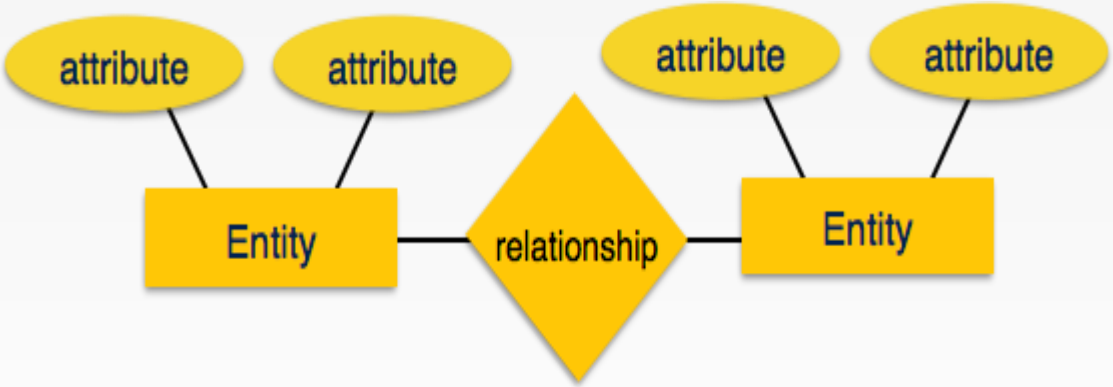
- **Entity:** An entity is referred to as a real-world object. It can be a name, place, object, class, etc. These are represented by a rectangle in an ER Diagram. E.g: Student, Course, or Company
- **Attributes:** An attribute can be defined as the description of the entity. These are represented by Ellipse in an ER Diagram. It can be Age, Roll Number, or Marks for a Student. E.g: StudentID, CourseName, or EmployeeEmail.
- **Relationship:** Relationships are used to define relations among different entities. Diamonds and Rhombus are used to show Relationships. E.g: Student enrolls in a Course.

Relational Model: Entity-Relationship Model(ER Model) cont…

Symbols Used in ER Model

- ER Model is used to model the logical view of the system from a data perspective which consists of these symbols:

Figures	Symbols	Represents
Rectangle		Entities in ER Model
Ellipse		Attributes in ER Model
Diamond		Relationships among Entities
Line		Attributes to Entities and Entity Sets with Other Relationship Types
Double Ellipse		Multi-Valued Attributes
Double Rectangle		Weak Entity



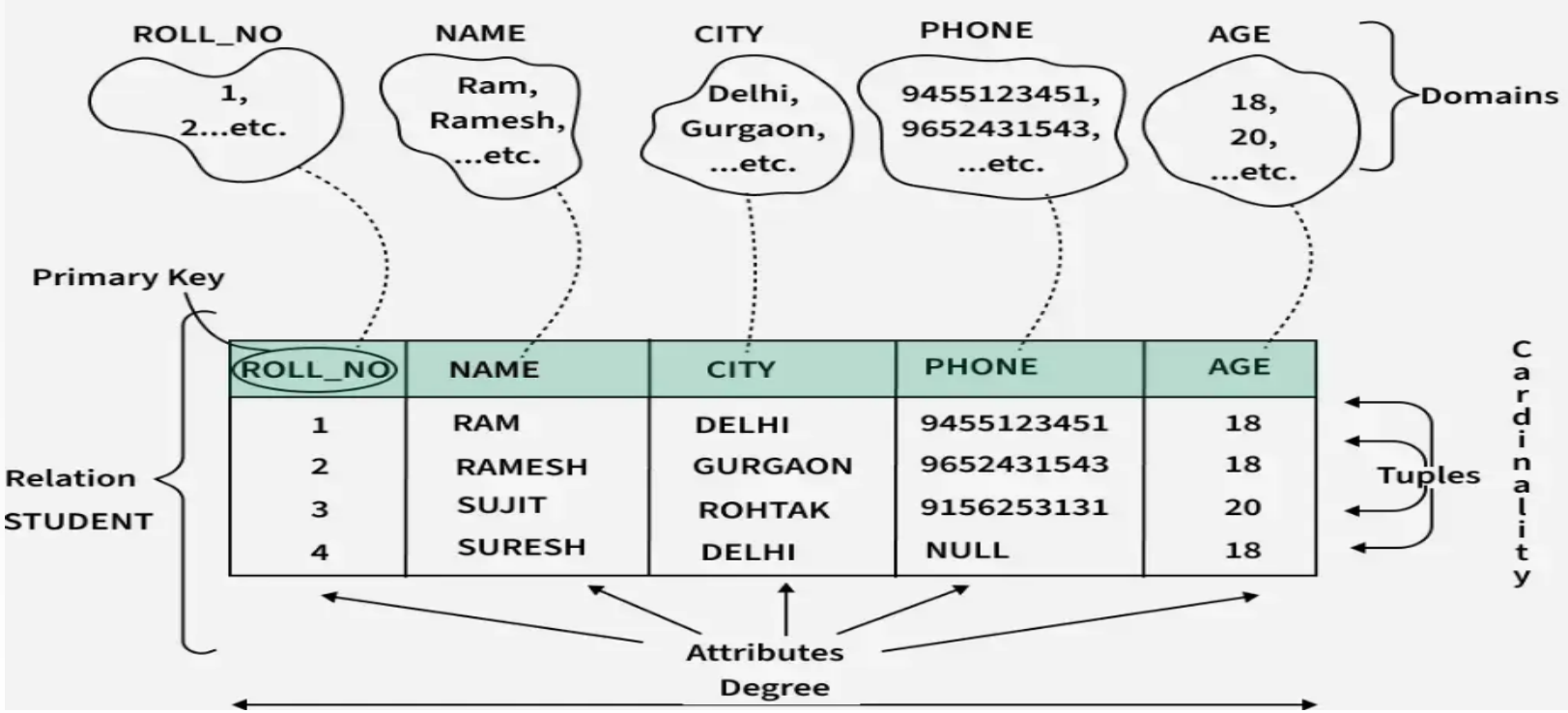
Key terms in Relational Model

- **Attribute:** Refer to the properties that define an entity. For example, ROLL_NO, NAME, ADDRESS etc.
- **Relation Schema:** It defines the structure of the relation and represents the name of the relation with its attributes. For example, STUDENT (ROLL_NO, NAME, ADDRESS, PHONE and AGE) is the relation schema for STUDENT. If a schema has more than 1 relation it is called Relational Schema.
- **Tuple:** It represents a row in a relation. Each tuple contains a set of attribute values that describe a particular entity. For example, (1, RAM, DELHI, 9455123451, 18) is a tuple in the STUDENT table.
- **Degree:** The number of attributes in the relation is known as the degree of the relation. For example, The STUDENT relation has a degree of 5, as it has 5 attributes.

Key terms in Relational Model

- **Cardinality:** The number of tuples in a relation is known as cardinality. For example, The STUDENT relation defined above has cardinality 4.
- **NULL Values:** The value which is not known or unavailable is called a NULL value. It is represented by NULL. For example, PHONE of STUDENT having ROLL_NO 4 is NULL.

Example >>> Consider a relation STUDENT with attributes ROLL_NO, NAME, ADDRESS, PHONE and AGE shown in the table.



Characteristics of the Relational Model

- **Atomic Values:** Each attribute in a table contains atomic values, meaning no multi-valued or nested data is allowed in a single cell.
- **Unique Keys:** Every table has a primary key to uniquely identify each record, ensuring no duplicate rows.
- **Attribute Domain:** Each attribute has a defined domain, specifying the valid data types and constraints for the values it can hold.
- **Data Independence:** The model ensures logical and physical data independence, allowing changes in the database schema without affecting the application layer.
- **Relational Operations:** Supports operations like selection, projection, join, union and intersection, enabling powerful data retrieval and manipulation.
- **Data Consistency:** Ensures data consistency through constraints, reducing redundancy and anomalies.

Types of Keys in the Relational Model

- **Primary Key:** A Primary Key uniquely identifies each tuple in a relation. It must contain unique values and cannot have NULL values.
- **Candidate Key:** A Candidate Key is a set of attributes that can uniquely identify a tuple in a relation.
- **Foreign Key:** A Foreign Key is an attribute in one relation that refers to the primary key of another relation.
- **Composite Key:** A Composite Key is formed by combining two or more attributes to uniquely identify a tuple.

Relationship in RDBMS

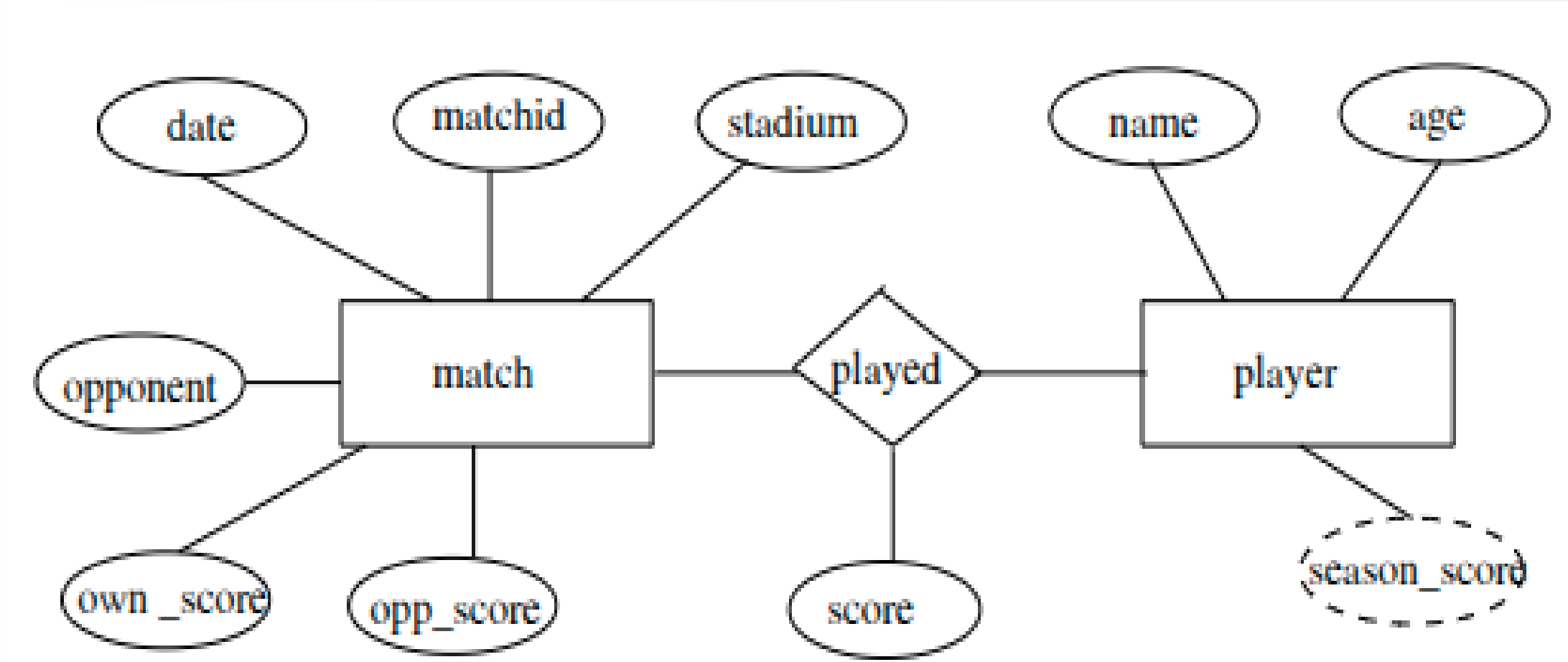
Relationships in SQL define how tables in a relational database are connected and interact through foreign keys, ensuring data integrity and enabling efficient data retrieval by allowing data to be linked across multiple tables.

Type of Relationships in SQL:

Exercise 1: ER Diagram

Design an E-R diagram for keeping track of the exploits of your favorite sports team. You should store the matches played, the scores in each match, the players in each match and individual player statistics for each match. Summary statistics should be modeled as derived attributes.

Exercise 1: ER Diagram



Thank you!!