

# DBMS Lec 6 (Relational Model):

The relational model is a way to organize and manage data using tables, where each table represents a set of related information and each row is a unique record. Keys like primary, foreign, and composite keys ensure data integrity, prevent duplication, and help link tables together, while integrity constraints enforce rules to keep the data accurate and consistent.

## Introduction to the Relational Model

The relational model is a type of data model used in database management systems to structure data using tables (relations). Each table consists of rows (records) and columns (attributes), making it easy to organize and retrieve information.

## Structure of Tables and Relations

In the relational model, each table represents an entity (like a customer), with columns for attributes (such as ID, name, address, contact) and rows for individual records. Each row is a unique instance of the entity, and all rows together form the relation.

## Understanding Cardinality and Degree

Cardinality refers to the number of rows (records) in a table, while degree refers to the number of columns (attributes). These concepts help describe the structure and size of a table.

## Converting Models to Relational Tables

Data models, like those used for an online delivery system, are first designed conceptually and then converted into relational tables. Each entity becomes a table, and its properties become columns.

## Properties of Relations and Tables

Each table must have a unique name, and each record is uniquely identified within the table. No two rows should be identical, and certain constraints ensure this uniqueness.

## Keys in the Relational Model

Keys are special attributes or combinations used to identify records uniquely. Types include super keys (any set of attributes that uniquely identify a row), candidate keys (minimal super keys), primary keys (chosen

candidate key), and alternate keys (other candidate keys not chosen as primary).

## Foreign Keys and Table Relationships

A foreign key is an attribute in one table that refers to the primary key of another table, creating a link between the two. This allows data to be related across tables, such as linking orders to customers.

## Composite Keys

A composite key uses two or more columns together to uniquely identify a record when a single column is not enough. This is useful when no single attribute can serve as a unique identifier.

## Integrity Constraints Overview

Integrity constraints are rules that ensure data in the database remains correct and consistent. These include entity integrity (primary keys cannot be null), referential integrity (foreign keys must match valid records), and domain constraints (values must be valid for the attribute type).

## **CRUD Operations and Data Consistency**

CRUD operations (Create, Read, Update, Delete) are basic actions on data. Integrity constraints are checked during these operations to maintain consistency and prevent invalid or duplicate data.

## **Referential Integrity and Foreign Key Constraints**

Referential integrity ensures that foreign key values in a child table must exist in the parent table. If a referenced record is deleted or updated, actions like cascading delete or setting null are used to maintain consistency.

## **Other Important Constraints: Not Null, Unique, Default, Check**

The Not Null constraint requires a column to always have a value. The Unique constraint ensures all values in a column are different. The Default constraint sets a standard value if none is provided. The Check constraint enforces a condition for valid data entry.

## **Primary Key Constraint**

The primary key uniquely identifies each record in a table and cannot be null. Only one primary key exists

per table, ensuring each record is distinct.

## **Foreign Key Constraint**

The foreign key constraint creates a relationship between two tables, ensuring that the value in the child table matches a valid value in the parent table's primary key. This maintains consistency across related data.

## **Summary and Next Steps**

The relational model uses tables, keys, and constraints to organize and protect data integrity. Understanding these concepts is essential for designing reliable databases. Future lessons will cover converting conceptual models into relational tables with examples.