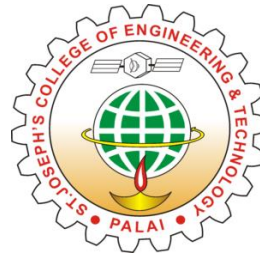


ST. JOSEPH'S
COLLEGE OF ENGINEERING
AND TECHNOLOGY,
- PALAI -



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CST 204 - Database Management Systems

Prof. Sarju S

10 May 2021

CST 204 – Database Management Systems

Module 2



- ▶ Structure of Relational Databases - Integrity Constraints, Synthesizing ER diagram to relational schema
- ▶ Introduction to Relational Algebra - select, project, cartesian product operations, join - Equi-join, natural join. query examples, introduction to Structured Query Language (SQL), Data Definition Language (DDL), Table definitions and operations – CREATE, DROP, ALTER, INSERT, DELETE, UPDATE.

Relational Model (RM)

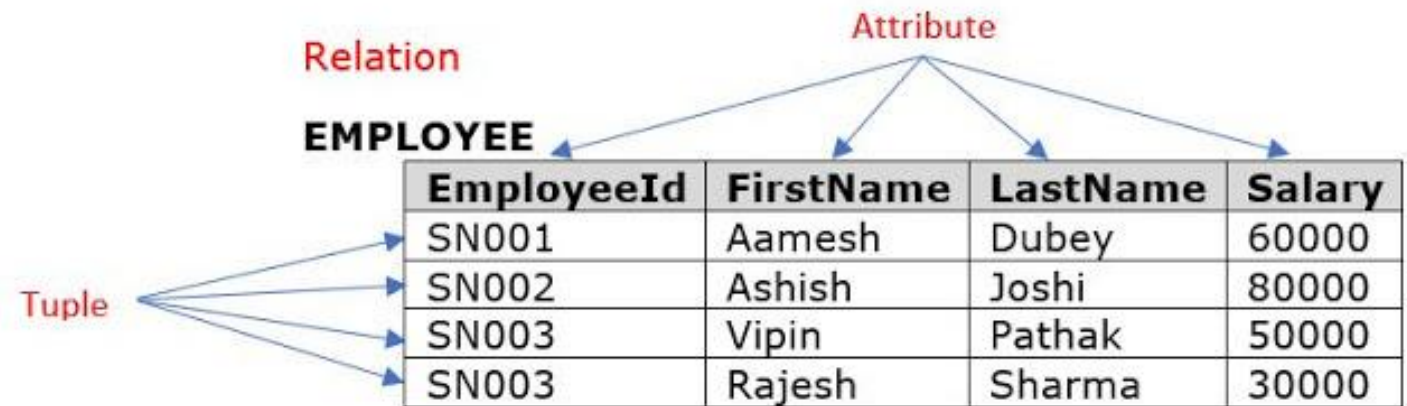
Relational Model (RM)



- ▶ Relational Model (RM) represents the database as a collection of relations.
- ▶ A relation is nothing but a table of values.
- ▶ Every row in the table represents a collection of related data values. These rows in the table denote a real-world entity or relationship.
- ▶ It is used for data storage and processing

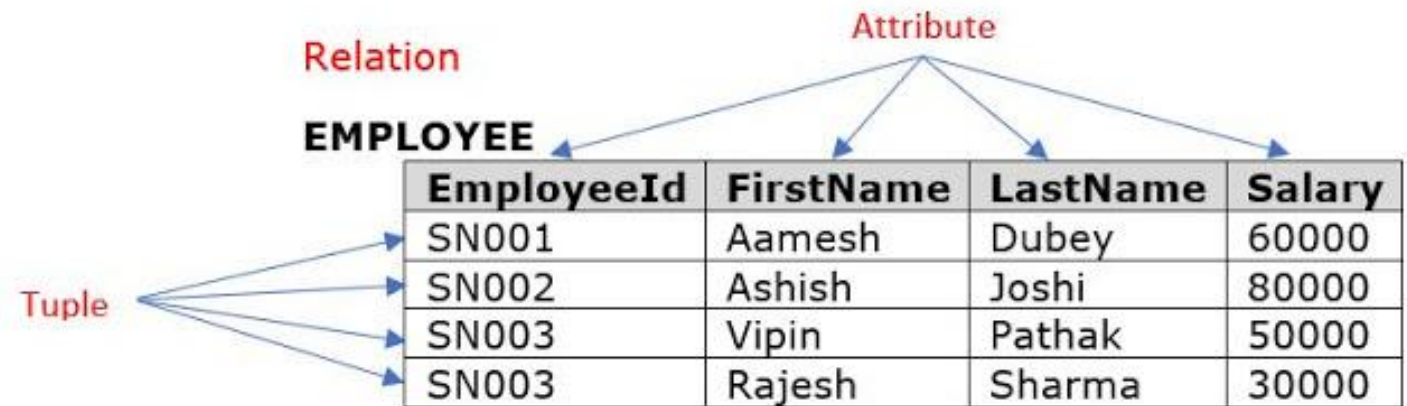
Relational Model (RM)

- ▶ **Attribute**: A column header of a table is known as attribute of a relation.
- ▶ **Tuple**: A row in a table represents the record of a relation and known as a tuple of a relation.
- ▶ **Domain**: A set of possible values for a given attribute is known as domain of a relation.
- ▶ **Relation Schema**: A relation schema represents name of the relation with its attributes. e.g.; EMPLOYEE (EmployeeId, FirstName, LastName, Salary)



Relational Model (RM)

- ▶ **Degree**: The number of attributes in the relation is known as degree of the relation. The EMPLOYEE relation defined above has degree 4.
- ▶ **Cardinality**: The number of tuples in a relation is known as cardinality. The EMPLOYEE relation defined has cardinality 4.



Relational Model (RM)



Informal Terms	Formal Terms
Table	Relation
Column header / field	Attributes
All possible column values	Domain
Row	Tuple
Table definition	Schema of relation

Characteristics of Relational Database Model



- ▶ Each **relation** in a database must have a **distinct or unique name** which would separate it from the other relations in a database.
- ▶ A relation must not have **two attributes** with the **same name**. Each attribute must have a distinct name.
- ▶ Duplicate tuples must not be present in a relation.

Roll_No.	Name	Department
101	Steive	Comp. Sci.
265	Jhoson	Finance
505	Margret	Biology
505	Margret	Biology
256	Davis	Comp. Sci.
453	Sheryl	Biology
365	Emma	Maths



Student Relation in Relational Model

Characteristics of Relational Database Model



- ▶ Each tuple must have exactly **one data value** for an **attribute**.
 - ▶ For example, below in the first table, you can see that for Roll_No. 265 we have enrolled two students Jhoson and Charles, this would not work. We must have only one student for one Roll_No.

Roll_No.	Name	Department	Roll_No.	Name	Department
101	Steive	Comp. Sci.	101	Steive	Comp. Sci.
265	Jhoson	Finance	265	Jhoson	Finance
	Charles		346	Charles	Finance
505	Margret	Biology	505	Margret	Biology
325	Jenny	Social Sci.	325	Jenny	Social Sci.
256	Davis	Comp. Sci.	256	Davis	Comp. Sci.
453	Sheryl	Biology	453	Sheryl	Biology
365	Emma	Maths	365	Emma	Maths

Characteristics of Relational Database Model



- ▶ Tuples in a relation **do not have** to follow a **significant order** as the relation is not order-sensitive.
- ▶ Similarly, the attributes of a relation also do not have to follow certain ordering, it's up to the developer to decide the ordering of attributes.

Relational Integrity Constraints

Relational Integrity Constraints



- ▶ Relational Integrity constraints in DBMS are referred to conditions which must be present for a valid relation.
- ▶ Constraints on the Relational database management system is mostly divided into three main categories are:
 - ▶ Domain Constraints
 - ▶ Entity integrity constraints
 - ▶ Key Constraints
 - ▶ Referential Integrity Constraints

Domain Constraints



- ▶ Each attribute in a tuple is declared to be of a particular domain (for example, integer, character, Boolean, String, etc.) which specifies a constraint on the values that an attribute can take.

ID	NAME	SEMENSTER	AGE
1000	Tom	1 st	17
1001	Johnson	2 nd	24
1002	Leonardo	5 th	21
1003	Kate	3 rd	19
1004	Morgan	8 th	A

Not allowed. Because AGE is an integer attribute

Entity integrity constraints



- ▶ The entity integrity constraint states that primary key value can't be null.
 - ▶ This is because the primary key value is used to identify individual rows in relation and if the primary key has a null value, then we can't identify those rows.
- ▶ A table can contain a null value other than the primary key field.

EMPLOYEE

EMP_ID	EMP_NAME	SALARY
123	Jack	30000
142	Harry	60000
164	John	20000
	Jackson	27000

Not allowed as primary key can't contain a NULL value

Key constraints



- ▶ Keys are the entity set that is used to identify an entity within its entity set uniquely.
- ▶ An entity set can have multiple keys, but out of which one key will be the primary key. A primary key can contain a unique and null value in the relational table.

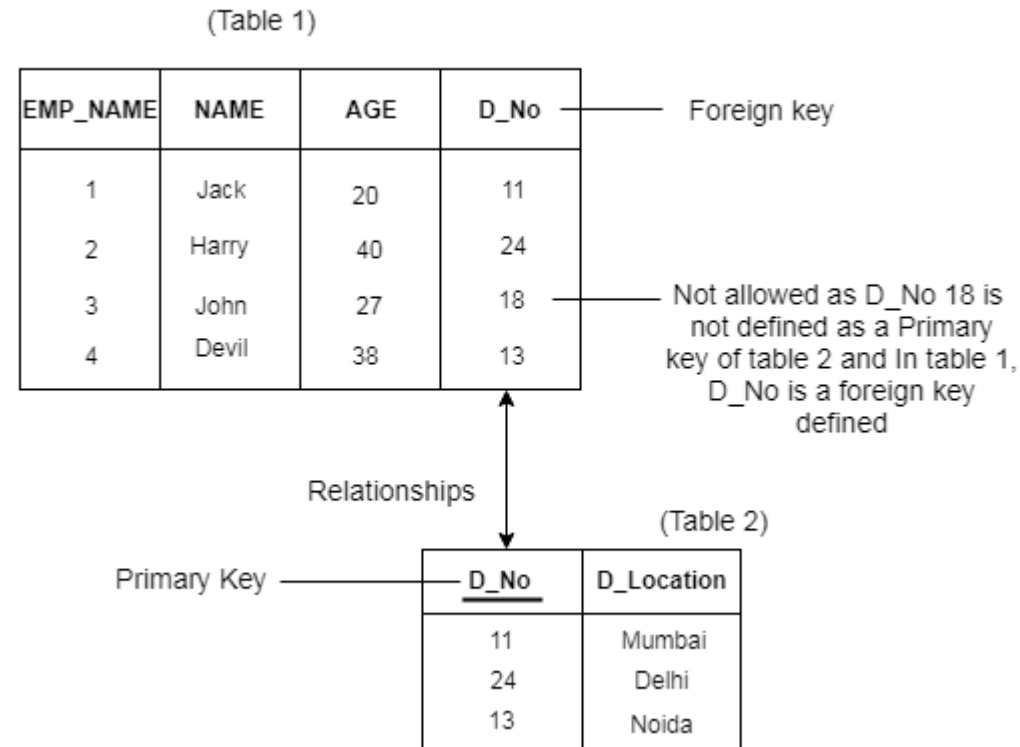
ID	NAME	SEMENSTER	AGE
1000	Tom	1 st	17
1001	Johnson	2 nd	24
1002	Leonardo	5 th	21
1003	Kate	3 rd	19
1002	Morgan	8 th	22

Not allowed. Because all row must be unique

Referential Integrity Constraints



- ▶ A referential integrity constraint is specified between two tables.
 - ▶ In the Referential integrity constraints, if a foreign key in Table 1 refers to the Primary Key of Table 2, then every value of the Foreign Key in Table 1 must be null or be available in Table 2.





Thank You



Prof. Sarju S

Department of Computer Science and Engineering
St. Joseph's College of Engineering and Technology, Palai
sarju.s@sjcetpalai.ac.in