Unit III – Database Design

Dependencies

Dependencies in DBMS is a relation between two or more attributes.

It has the following types in DBMS

- Functional Dependency
- Fully-Functional Dependency
- Transitive Dependency
- Multivalued Dependency
- Partial Dependency

Normal Forms

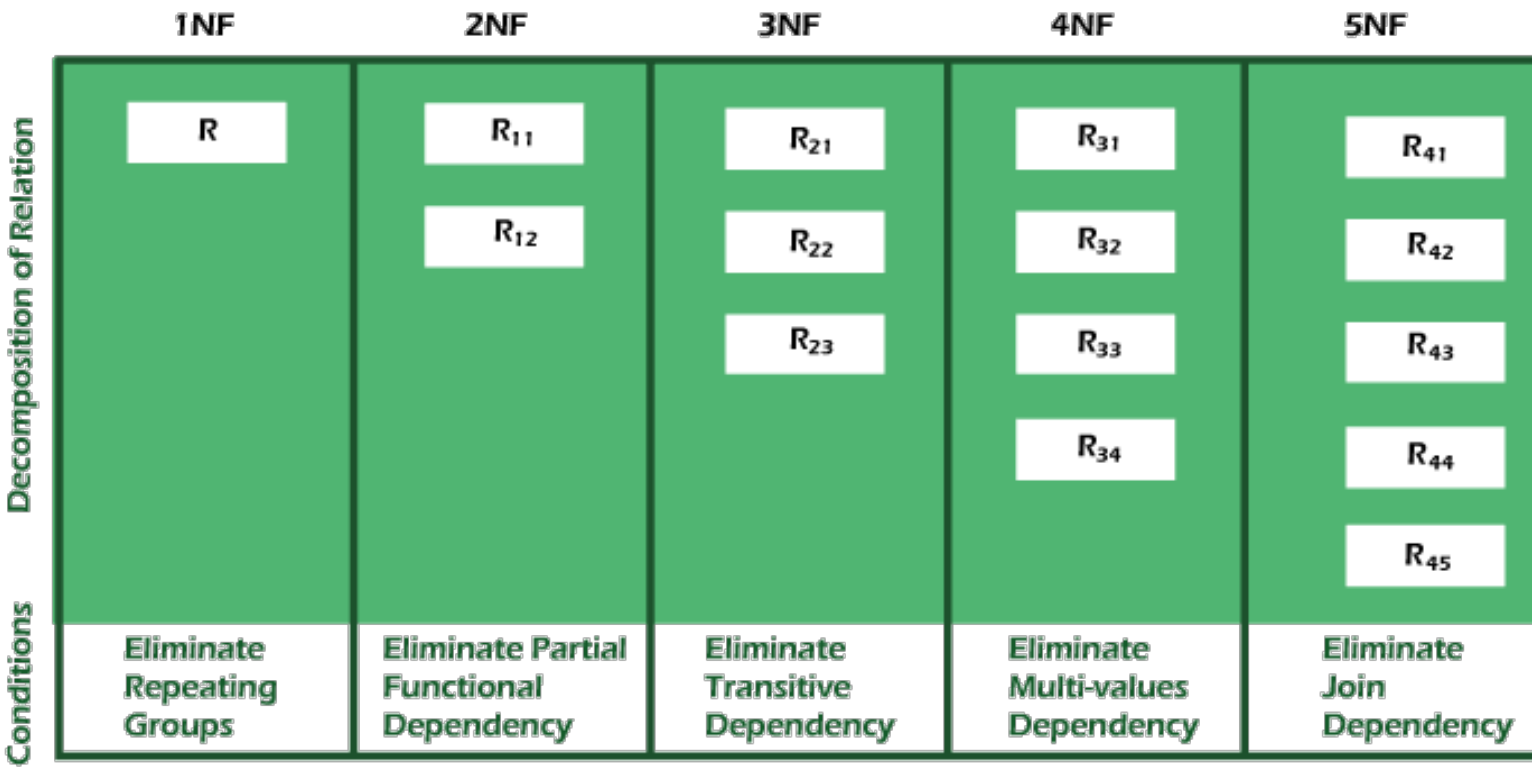
Normalization is the process of minimizing **redundancy** from a relation or set of relations.

Redundancy in relation may cause insertion, deletion, and update anomalies.

So, it helps to minimize the redundancy in relations.

Normal forms are used to eliminate or reduce redundancy in database tables.

- First Normal Form
- Second Normal Form
- Third Normal Form
- Boyce Codd Normal Form
- Fourth Normal Form
- Fifth Normal Form



Functional Dependencies

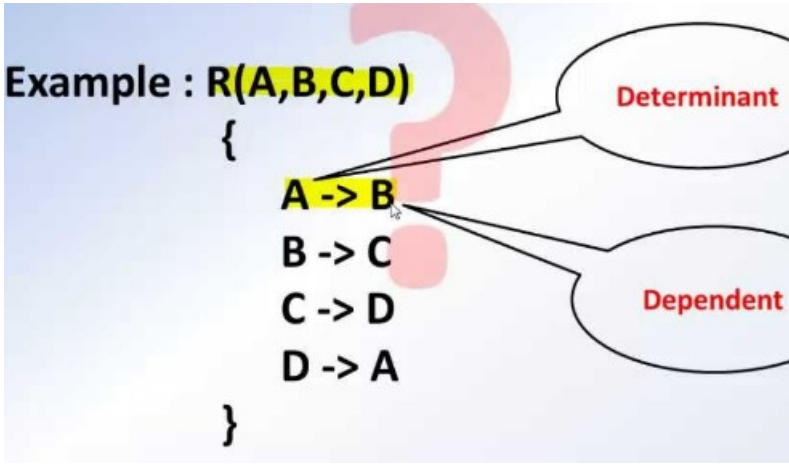
A functional dependency is a constraint that specifies the relationship between two sets of attributes

- where one set can accurately determine the value of other sets.

It is denoted as **X → Y**,

where X is a set of attributes that is capable of determining the value of Y.

The attribute set on the left side of the arrow, **X** is called **Determinant**, while on the right side, **Y** is called the **Dependent**.



Example 1

oll_no	Name	Marks	Dept	Course
1	A	78	CS	C1
2	B	60	EE	C1
3	A	78	CS	C2
4	B	60	EE	C3
5	C	80	IT	C3
6	d	80	EC	C2

Roll_no	Name
Name	Roll_no
Roll_no	marks
Dept	Course
Course	Dept
Roll_no,Name	Marks
Name	Marks
Name, Marks	Dept
Name, Marks	Dept, Course
Roll_no	Name, marks
Dept, Course	Name
Roll_no,Marks	Dept
Name	Course
Name,Marks, Dept	Roll_no

Example 2

roll_no	name	dept_name	dept_building
42	abc	CO	A4
43	pqr	IT	A3
44	xyz	CO	A4
45	xyz	IT	A3
46	mno	EC	B2
47	jkl	ME	B2

Valid Functional Dependencies

roll_no → { name, dept_name, dept_building },→ Here, roll_no can determine values of field name, dept_name and dept_building, hence a valid Functional dependency

roll_no → dept_name , Since, roll_no can determine whole set of {name, dept_name, dept_building}, it can determine its subset dept_name also.

dept_name → dept_building , Dept_name can identify the dept_building accurately, since departments with different dept_name will also have a different dept_building

More valid functional dependencies: roll_no → name, {roll_no, name} ⇢ {dept_name, dept_building}, etc.

invalid functional dependencies

ame → dept_name Students with the same name can have different dept_name,

ence this is not a valid functional dependency.

ept_building → dept_name There can be multiple departments in the same buildi

or example, in the above table departments ME and EC are in the same building B2,

ence dept_building → dept_name is an invalid functional dependency.

lore invalid functional dependencies: name → roll_no, {name, dept_name} → roll_no

ept_building → roll_no, etc.


Thank You!