

Functional dependencies in DBMS

- 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 $X \rightarrow Y$, 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:

Employee number	Employee Name	Salary	City
1	Dana	50000	San Francisco
2	Francis	38000	London
3	Andrew	25000	Tokyo

Rules of Functional Dependencies

Below are the three most important rules for Functional Dependency in Database:

- Reflexive rule –. If X is a set of attributes and Y is_subset_of X , then X holds a value of Y .
- Augmentation rule: When $x \rightarrow y$ holds, and c is attribute set, then $ac \rightarrow bc$ also holds. That is adding attributes which do not change the basic dependencies.
- Transitivity rule: This rule is very much similar to the transitive rule in algebra if $x \rightarrow y$ holds and $y \rightarrow z$ holds, then $x \rightarrow z$ also holds. $X \rightarrow y$ is called as functionally that determines y .

Types of Functional Dependencies in DBMS

There are mainly four types of Functional Dependency in DBMS. Following are the types of Functional Dependencies in DBMS:

- Multivalued Dependency
- Trivial Functional Dependency
- Non-Trivial Functional Dependency
- Transitive Dependency

Multivalued Functional Dependency

In Multivalued functional dependency, entities of the dependent set are not dependent on each other.

i.e. If $a \rightarrow \{b, c\}$ and there exists **no functional dependency** between **b and c**, then it is called a multivalued functional dependency.

For example,

roll_no	name	age
42	abc	17
43	pqr	18
44	xyz	18
45	abc	19

Here, $\text{roll_no} \rightarrow \{\text{name, age}\}$ is a multivalued functional dependency, since the dependents **name** & **age** are **not dependent** on each other (i.e. **name** \rightarrow **age** or **age** \rightarrow **name** doesn't exist !)

Trivial Functional Dependency

In **Trivial Functional Dependency**, a dependent is always a subset of the determinant. i.e. If $X \rightarrow Y$ and **Y is the subset of X**, then it is called trivial functional dependency

For example,

roll_no	name	age
42	abc	17
43	pqr	18
44	xyz	18

Here, $\{\text{roll_no, name}\} \rightarrow \text{name}$ is a trivial functional dependency, since the dependent **name** is a subset of determinant set $\{\text{roll_no, name}\}$
Similarly, $\text{roll_no} \rightarrow \text{roll_no}$ is also an example of trivial functional dependency.

Non-trivial Functional Dependency

In **Non-trivial functional dependency**, the dependent is strictly not a subset of the determinant.

i.e. If $X \rightarrow Y$ and **Y is not a subset of X**, then it is called Non-trivial functional dependency.

For example,

roll_no	name	age
42	abc	17
43	pqr	18
44	xyz	18

Here, $\text{roll_no} \rightarrow \text{name}$ is a non-trivial functional dependency, since the dependent **name** is **not a subset of** determinant **roll_no**

Similarly, $\{\text{roll_no}, \text{name}\} \rightarrow \text{age}$ is also a non-trivial functional dependency, since **age** is **not a subset of** $\{\text{roll_no}, \text{name}\}$ _

Transitive Functional Dependency

In transitive functional dependency, dependent is indirectly dependent on determinant.

i.e. If $a \rightarrow b$ & $b \rightarrow c$, then according to axiom of transitivity, $a \rightarrow c$. This is a **transitive functional dependency**

For example,

enrol_no	name	dept	building_no
42	abc	CO	4
43	pqr	EC	2
44	xyz	IT	1

enrol_no	name	dept	building_no
45	abc	EC	2

Here, **enrol_no** → **dept** and **dept** → **building_no**,

Hence, according to the axiom of transitivity, **enrol_no** → **building_no** is a valid functional dependency. This is an indirect functional dependency, hence called Transitive functional dependency.