FUNCTIONAL DEPENDENCIES

Engr. Laraib Siddiqui

Functional Dependency

- Functional dependency (FD) is a set of constraints between two attributes in a relation.
- Functional dependency says that if two tuples have same values for attributes A1, A2,..., An, then those two tuples must have to have same values for attributes B1, B2, ..., Bn.
  - A $\longrightarrow$ B
- It does not mean that A derives B, although it may be the case sometime
- It means that if we know value of A then we can precisely determine a unique value of B
- Normalization is based on functional dependencies (FDs)

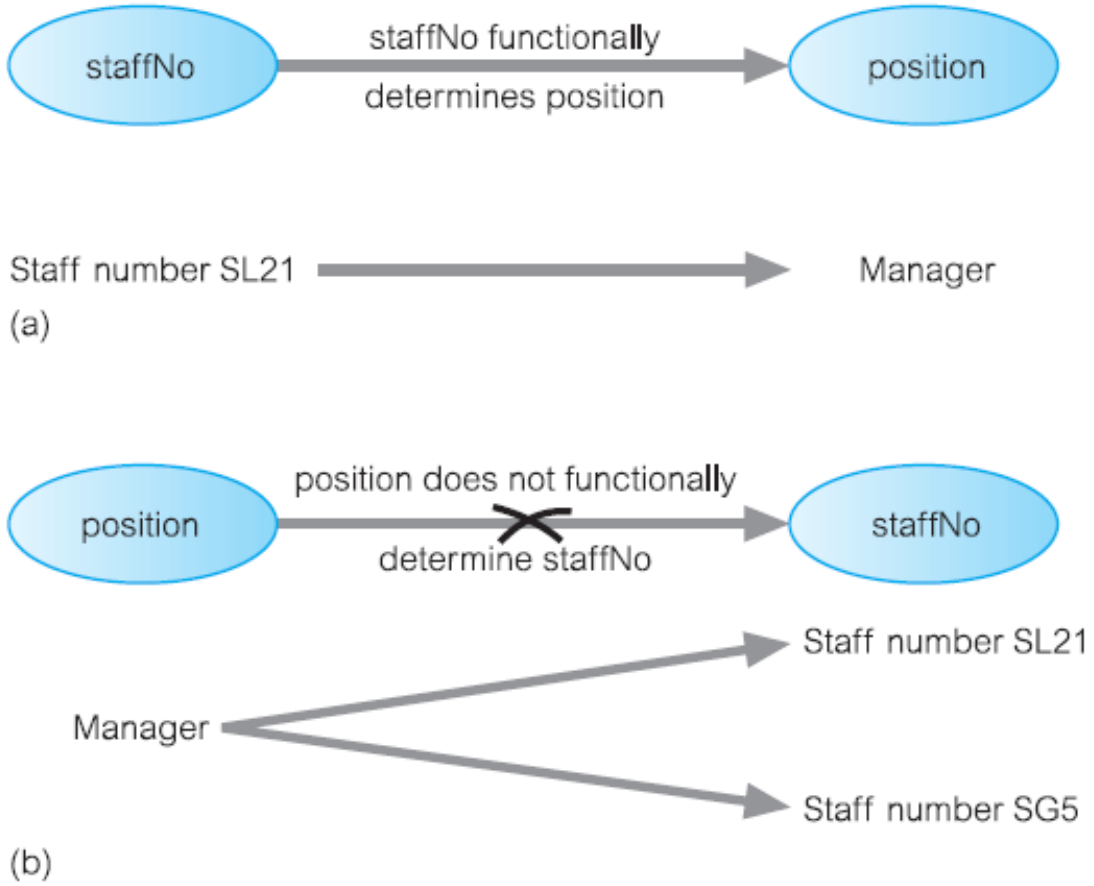
Functionally Dependency

- Attribute of set of attributes on the left side are called determinant and on the right are called dependents
- Like R (a, b, c, d, e)

  a → b, c, d

  d → d, e

Functional dependency example



Functional dependency

Fully Functional dependency

Indicates that if A and B are attributes of a relation, B is fully functionally dependent on A if B is functionally dependent on A, but not on any proper subset of A.

Partial dependency

Indicates that if there is some attribute that can be removed from A and yet the dependency still holds.

EMP(eId, eName, eAdr, eDept, prId, prSal)

eId → eName, eDept, eAdr

eId, prId → prSal

Inference Rules

- Called inference axioms or Armstrong axioms
- These are rules that establish certain FDs from a given set of FDs

Reflexivity

If B is a subset of A then A⟶B, it also implies that A always hold A, that is

stName, stAdr⟶ stName

Or stName⟶stName

Augmentation

If we have A⟶B then

AC⟶BC that is

if stId⟶stName then

stId, stAdr⟶ stName, stAdr

Transitivity

If A⟶B and B⟶C then A⟶C

that is

If stId⟶prName and prName⟶credits

Then

stId ⟶ credits

Additivity or Union

If A⟶B and A⟶C then A⟶BC

if empId ⟶eName and empId⟶qual

Then we can write it as

empId⟶ eName, qual

Projectivity or Decomposition

If A⟶BC then A⟶B and A⟶C

if empId⟶ eName, qual

Then we can write it as

empId ⟶eName and empId ⟶qual

Pseudo transitivity

If A → B and CB → D then AC → D

if stId → stName and

stName, fName → stAdr

Then we can write it as

stId, fName → stAdr

Types of FDs

1. Trivial functional dependency
2. Non-Trivial functional dependency
3. Multivalued functional dependency
4. Transitive functional dependency

Trivial functional dependency

A dependent is always a subset of the determinant.

If $X \rightarrow Y$ and Y is the subset of X, then it is called trivial functional dependency

{roll_no, name} → **name** is a trivial functional dependency, since the dependent name is a subset of determinant set **{roll_no, name}**
Similarly, **roll_no** → **roll_no** is also an example of trivial functional dependency.

Roll_no	Name	Age
12	Ali	17
13	Osama	18
14	Faraz	18

Non-Trivial functional dependency

The dependent is strictly not a subset of the determinant.

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

roll_no → **name** is a non-trivial functional dependency, since the dependent name is not a subset of determinant **roll_no**
Similarly, **{roll_no, name}** → **age** is also a non-trivial functional dependency, since age is not a subset of **{roll_no, name}**

Roll_no	Name	Age
12	Ali	17
13	Osama	18
14	Faraz	18

Multivalued functional dependency

Entities of the dependent set are not dependent on each other.

If a ⟶ {b, c} and there exists no functional dependency between b and c, then it is called a multivalued functional dependency.

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

Roll_no	Name	Age
12	Ali	17
13	Osama	18
14	Faraz	18

Transitive functional dependency

Dependent is indirectly dependent on determinant.

If a → b & b → c, then according to axiom of transitivity, a → c. This is a transitive functional dependency

Here, **Roll_no → dept** and **dept → buildingno,** Hence, according to the axiom of transitivity,**Roll_no → buildingno** is a valid functional dependency. This is an indirect functional dependency, hence called Transitive functional dependency.

Roll_no	name	dept	buildingno
12	Ali	SE	4
13	Osama	EE	2
14	Faraz	CS	1
15	Sara	EE	2