What is a functional dependency?

To define this in simpler terms, it's a relationship between the primary key attribute (usually) of the Relation to that of the other attributes in that relation.

Suppose, M is the Primary key attribute of the Relation XYZ. Now there's another attribute named N. Now let's suppose that attribute 'N' has certain data entries with the same values and to distinguish among them or you say to accurately determine the values in the attribute 'N' we take help of the Primary key attribute 'M' and we determine values for attribute 'N' through it.

Now this is called a functional dependency of 'N' on 'M'.

We usually represent Functional dependency with an arrow symbol. As shown below:

$\longrightarrow$

To represent the above stated situation with the symbol, it will be like,

$M \longrightarrow N$

Here, we call 'M' the determinant and 'N' the dependent.

Note: Usually the determinant is the Primary key of the relation, but there could exist the case where a certain set of attributes can be determined out of some attribute but that attribute isn't the primary key attribute.

Let's take a small example to visualise functional dependencies better,

C1	C2
7	21
3	100
7	5
5	21

Here, if we have to consider the permutation and combinations of all FD's relations (whether possible or not) we can have 2 relations that is

C1$\longrightarrow$ C2

C2⟶ C1

Now on this instance,

- If we observe the values carefully in the Relation above it's quite safe to say that **C1⟶ C2 does NOT hold on the above relation** because '7' appears in C1 twice and due to that we won't be able to determine correct corresponding values in C2. Hence we can't determine C2 from C1.

- Whereas, **C2⟶ C1 does hold on the relation,** as C1 uniquely identifies each value in C2.

We can also define FD's as the legal relations of the given instance of table/relation. Like in the above example, C2⟶ C1 is a legal relation while C1⟶ C2 is an illegal one.

Why do we need Functional Dependencies ?

It helps us to maintain the quality of the data in the database. As we have already familiarised with the notion that Functional Dependencies is a relationship between columns/attributes of a Relation/Table dependent on each other.
How does it do that?

- Functional dependencies check the legality of the relation tuple under a given set of FDs.
- It defines the constraints on the Relation.

Rules of Functional Dependencies:

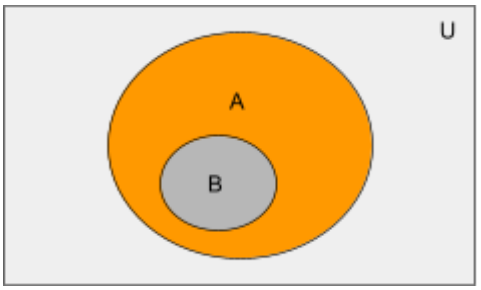
1. Reflexive Rule: If 'A' is a set of attributes and 'B' is a subset of 'A'. Then, A⟶ B holds.
2. Augmentation Rule: If B can be determined from A, then adding an attribute to this functional dependency won't change anything.
   i.e. If A⟶ B holds, then AM⟶ BM holds too. 'M' being a set of attributes.
3. Transitivity Rule: if A determines B and B determines C, we can say that A determines C.
   i.e. if A⟶ B and B⟶ C then A⟶ C

   Note: These rules are also known as Armstrong's axioms.

Types of Functional Dependencies:

1. **Trivial Functional Dependency:** Let there be set of attributes A which determines set of attributes B (i.e. A⟶ B).
   Now if the dependent (i.e. B) is a subset of the determinant (i.e. A).
   i.e. If A⟶ B and B⊆A
   Hence, we call this a trivial functional dependency.

   Venn Diagram for better understanding:



   Note: A set is also a subset of its own, i.e. A⊆A. Hence, A⟶ A is also a trivial functional dependency.

For Example:

owner_id	owner_name
2352352	Ted Mosby
3523523	Ross Geller
7485685	Joey Tribbiani

Consider the above given table with attributes named owner_id and owner_name.
Here, {owner_id,owner_name} ⟶ owner_name is a trivial functional dependency since owner_name is a subset of {owner_id,owner_name}.
owner_id⟶ owner_id is also a trivial functional dependency.

2. **Non-trivial Functional Dependency:** Let there be set of attributes A which determines set of attributes B (i.e. A⟶ B).
   Now if the dependent (i.e. B) is **NOT** a subset of the determinant (i.e. A).
   Hence, we call this a Non-trivial functional dependency.

For Example:

owner_id	owner_name	cars_owned
2352352	Ted Mosby	Pontiac Fiero
3523523	Ross Geller	MG model B
7485685	Joey Tribbiani	Porsche

Consider the above given table with attributes named owner_id, owner_name and cars_owned.

Here, owner_id⟶ owner_name , {owner_id,owner_name} ⟶ cars_owned are examples of non-trivial functional dependencies as their dependent isn't a subset of its respective determinant.