Introduction to Normalization

Super Key: A key that uniquely identify each row. A relation might have many Super Keys.

Candidate Key: A super key without unnecessary attributes. In another words, it is minimal Super Key. We may have more than one Candidate Keys

Primary Key: One of the Candidate Keys.

Functional Dependency: The relationship between the attributes of a relation (table).

Example: R(a, b, c, d)

Functional Dependency:

1. a,b → c, d (a and b determine c and d)
2. c→d (c determines d) or (d is functionally dependent on c)

Determinant: left hand-side of the functional dependency is known as determinant.

What is Normalization?

The normalization is a procedure that eliminates data redundancy while ensuring data integrity.

Data integrity:

- **Entity Integrity:** Every relation (table) must have a Primary Key attribute/column. That means each table must be uniquely identified by one or more than one columns in a relation.
- **Referential Integrity:** Indicates the Foreign Keys. The relationship between the entities. We have to make sure that the links between the tables are CORRECT.

In a proper Database Design the phases should be in the following order:

- Phase 1: Analyzing the problem/scenario – identify all the database requirements (possible entities, their attributes, relationships between the entities and other business rules)
- Phase 2: Drawing ERD using all database/system requirements that are listed in Phase 1.
- Phase 3: Relational Model – converting ERD to Relational Schema- which should include Primary Keys as well as Foreign Keys.
- Phase 4: Normalize tables if necessary (if data redundancy exist)

If the phases of Database Design are not applied in a correct order, we might have redundancy.

Some programmers, do not want to deal with many tables (pks, fks), they prefer to work with a flat table. This table might include all attributes that should be kept in the database together.

Ex: Assume for hospital database we keep all necessary data (for patients, doctors, nurses, treatments, drugs, etc.) in a single table.

We will definitely have data redundancy in this kind of cases!!! Then we need to normalize the database/tables.