**Normalization in Relational Database**

Normalization is the process of organizing data in a relational database to minimize redundancy and dependency by dividing the data into well-structured tables. The goal of normalization is to ensure the data is stored logically, efficiently, and consistently, reducing anomalies during data operations (insert, update, and delete).

**Key Concepts in Normalization:**

1. **Data Redundancy:** Eliminate duplicate data.
2. **Data Dependency:** Ensure logical dependencies are preserved.
3. **Anomaly Prevention:** Avoid anomalies like insertion, update, and deletion anomalies.

**Forms of Normalization:**

1. **First Normal Form (1NF):**
   * Eliminate repeating groups or arrays in a table.
   * Ensure each column contains atomic (indivisible) values.
2. **Second Normal Form (2NF):**
   * Meet all 1NF requirements.
   * Remove partial dependencies (i.e., no non-prime attribute depends on a part of the primary key).
3. **Third Normal Form (3NF):**
   * Meet all 2NF requirements.
   * Eliminate transitive dependencies (i.e., non-prime attributes depend on other non-prime attributes).
4. **Boyce-Codd Normal Form (BCNF):**
   * A stricter version of 3NF.
   * Ensure every determinant is a candidate key.
5. **Fourth Normal Form (4NF) and Fifth Normal Form (5NF):**
   * Deal with multi-valued dependencies and join dependencies, respectively.

Normalization simplifies database design, promotes data integrity, and enhances query performance but can sometimes lead to performance trade-offs due to the need for complex joins.

**Entity-Relationship (ER) Diagrams**

ER diagrams are a visual representation of the entities, relationships, and attributes of a database system. They serve as a blueprint for designing relational databases.

**Components of ER Diagrams:**

1. **Entities:**
   * Represent objects or concepts with distinct existence.
   * Depicted as rectangles.
   * Types:
     + **Strong Entity:** Can exist independently.
     + **Weak Entity:** Depends on a strong entity and has a discriminator.
2. **Attributes:**
   * Describe properties of an entity or relationship.
   * Depicted as ovals.
   * Types:
     + **Simple Attribute:** Cannot be divided further (e.g., Name).
     + **Composite Attribute:** Can be divided into sub-parts (e.g., Full Name = First Name + Last Name).
     + **Derived Attribute:** Derived from other attributes (e.g., Age from Date of Birth).
3. **Relationships:**
   * Represent associations between entities.
   * Depicted as diamonds.
   * Types:
     + **One-to-One (1:1):** An entity in set A is related to one entity in set B and vice versa.
     + **One-to-Many (1:N):** An entity in set A is related to multiple entities in set B, but not vice versa.
     + **Many-to-Many (M:N):** Entities in set A are related to multiple entities in set B and vice versa.
4. **Cardinality:**
   * Specifies the number of instances of an entity associated with one instance of another entity.
   * Represented as numbers or symbols near the relationship line.
5. **Primary and Foreign Keys:**
   * **Primary Key:** A unique identifier for an entity.
   * **Foreign Key:** A key in one table that refers to the primary key in another table to establish relationships.

ER diagrams are essential for visualizing and designing normalized databases, providing a high-level understanding of database structure before implementation.