Normalization Process

1. Unnormalized Form (UNF):

Initially, the data might have been stored in a single large table with redundancy and no structure. For example:

Email Address	Roll Number	Full Name	Gender	Contact Number	Programme	Specialization1	Fathers Name	Fathers Occupation	Certification	Certification Source	Leadership Position
student1@example.com	12345	Alice Smith	Female	555-1234	MBA	Marketing	John Smith	Engineer	Project Management (PMP)	PMI	Class Representative
student2@example.com	12346	Bob Johnson	Male	555-5678	MBA	Finance	Mark Johnson	Doctor	Certified Data Analyst	Coursera	Club President

Issues in UNF:

- Redundancy: Fathers' details, certifications, and leadership positions are repeated for each student.

- Atomicity: Multiple values, like certifications and leadership positions, could be stored in a single column.

2. First Normal Form (1NF):

In 1NF, the table is restructured to ensure that all values are atomic and the data has a primary key.

Example: Students Table in 1NF

Email Address	Roll Number	Full Name	Gender	Contact Number	Programme	Specialization1	Fathers Name	Fathers Occupation
student1@example.com	12345	Alice Smith	Female	555-1234	MBA	Marketing	John Smith	Engineer
student2@example.com	12346	Bob Johnson	Male	555-5678	MBA	Finance	Mark Johnson	Doctor

- Primary Key: `EmailAddress` ensures that each record is uniquely identifiable.

- Atomic Values: Each column contains only one value per record.

3. Second Normal Form (2NF):

In 2NF, we eliminate partial dependencies. All non-key attributes should depend on the entire primary key.

Example: Normalizing `ParentDetails` into a Separate Table

Original Structure (1NF):

Email Address	Fathers Name	Fathers Occupation	Mothers Name	Mothers Occupation

student1@example.com	John Smith	Engineer	Mary Smith	Teacher
student2@example.com	Mark Johnson	Doctor	Sarah Johnson	Nurse

2NF Normalized Structure:

- `Students` Table:

Email Address	Roll Number	Full Name	Gender	Contact Number
student1@example.com	12345	Alice Smith	Female	555-1234
student2@example.com	12346	Bob Johnson	Male	555-5678

- `ParentDetails` Table:

ParentID	Email Address	Parent Type	Parent Name	Occupation	Designation	Contact Number
1	student1@example.com	Father	John Smith	Engineer	Manager	555-1111
2	student1@example.com	Mother	Mary Smith	Teacher	Principal	555-2222
3	student2@example.com	Father	Mark Johnson	Doctor	Surgeon	555-3333

Explanation:

- Each parent is now stored separately, linked by `EmailAddress` to the corresponding student.

4. Third Normal Form (3NF):

In 3NF, we eliminate transitive dependencies. Non-key attributes should not depend on other non-key attributes.

Example: `Certifications` and `LeadershipPositions` Tables

Original Structure (1NF):

Email Address	Certification	Certification Source	Leadership Position
student1@example.com	Project Management (PMP)	PMI	Class Representative
student2@example.com	Certified Data Analyst	Coursera	Club President

3NF Normalized Structure:

- `Certifications` Table:

CertificationID	Email Address	Certification	Certification Source
1	student1@example.com	Project Management (PMP)	PMI
2	student2@example.com	Certified Data Analyst	Coursera

- `LeadershipPositions` Table:

LeadershipID	Email Address	Position
1	student1@example.com	Class Representative
2	student2@example.com	Club President

Explanation:

- Separate tables for certifications and leadership positions remove redundancy and ensure that these attributes are linked to students without unnecessary repetition.

Final Structure and Relations

After applying all normalization steps, the database structure is now well-organized, minimizing redundancy and ensuring data integrity.

- Students Table links to:
  - ParentDetails (via `EmailAddress`)
  - Certifications (via `EmailAddress`)
  - LeadershipPositions (via `EmailAddress`)

Each table is normalized to 3NF, ensuring that all non-key attributes are fully and directly dependent on the primary key, with no partial or transitive dependencies.

Summary:

- 1NF: Ensured atomicity and established primary keys.

- 2NF: Eliminated partial dependencies by separating parent details.

- 3NF: Removed transitive dependencies by creating separate tables for certifications and leadership positions, with appropriate foreign key relationships.