Day - 8 Assignments **Ankit Kumar Keshri**

Assignment 1: Analyze a given business scenario and create an ER diagram that includes entities, relationships, attributes, and cardinality. Ensure that the diagram reflects proper normalization up to the third normal form.

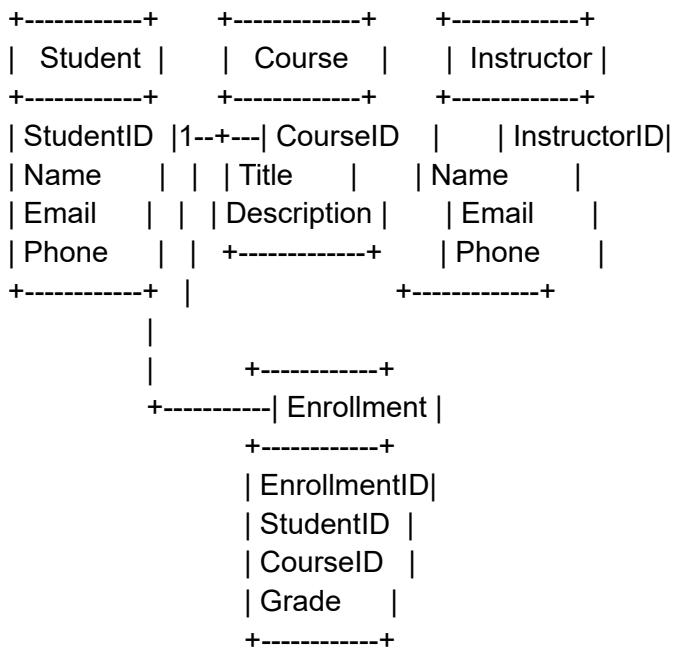
A university wants to design a database system to manage its student enrollment, courses, and instructors.

Entities:

1. Student:
    - Attributes: StudentID (Primary Key), Name, Email, Phone
2. Course:
    - Attributes: CourseID (Primary Key), Title, Description
3. Instructor:
    - Attributes: InstructorID (Primary Key), Name, Email, Phone
4. Enrollment:
    - Attributes: EnrollmentID (Primary Key), StudentID (Foreign Key), CourseID (Foreign Key), Grade

Relationships:

1. Enrolls In (between Student and Course):
    - Each student can enroll in multiple courses.
    - Each course can have multiple students enrolled.
    - Cardinality: Many-to-Many
2. Teaches (between Instructor and Course):
    - Each instructor can teach multiple courses.
    - Each course is taught by one instructor.
    - Cardinality: One-to-Many
3. Enrollment Details (between Student and Enrollment):
    - Each student can have multiple enrollments.
    - Each enrollment is associated with one student.
    - Cardinality: One-to-Many
4. Enrollment Details (between Course and Enrollment):
    - Each course can have multiple enrollments.
    - Each enrollment is associated with one course.
    - Cardinality: One-to-Many



Normalization:

- First Normal Form (1NF):
    - Each table has a primary key.
    - Each attribute contains atomic values.
- Second Normal Form (2NF):
    - No partial dependencies exist.
- Third Normal Form (3NF):
    - No transitive dependencies exist.

Assignment 2: Write SQL statements to CREATE a new database and tables that reflect the library schema you designed earlier. Use ALTER statements to modify the table structures and DROP statements to remove a redundant table.

-- Create the database

CREATE DATABASE library_db;

```
-- Use the database
USE library_db;

-- Create the tables
CREATE TABLE Student (
    StudentID INT PRIMARY KEY,
    Name VARCHAR(50),
    Email VARCHAR(100),
    Phone VARCHAR(20)
);

CREATE TABLE Course (
    CourseID INT PRIMARY KEY,
    Title VARCHAR(100),
    Description TEXT
);

CREATE TABLE Instructor (
    InstructorID INT PRIMARY KEY,
    Name VARCHAR(50),
    Email VARCHAR(100),
    Phone VARCHAR(20)
);
```

```
CREATE TABLE Enrollment (
    EnrollmentID INT PRIMARY KEY,
    StudentID INT,
    CourseID INT,
    Grade VARCHAR(2),
    FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
    FOREIGN KEY (CourseID) REFERENCES Course(CourseID)
);

-- Add a new column to the Course table
ALTER TABLE Course
ADD StartDate DATE;

-- Drop the redundant table
DROP TABLE `Enrollment Details`;
```