Assignment: DDL Commands – Student Database

Design a simple Student Database for a college. The database should store information about students, their courses, and enrollments.

Questions

**Q1. CREATE Tables (with & without constraints)**

***Create a table students with the following columns (add constraints where needed):***

***student\_id (INT, Primary Key)***

***first\_name (VARCHAR(50), NOT NULL)***

***last\_name (VARCHAR(50))***

***dob (DATE, NOT NULL)***

***gender (CHAR(1), check constraint: only 'M' or 'F')***

***Create another table courses without constraints with the following columns:***

***course\_id (INT)***

***course\_name (VARCHAR(100))***

***credits (INT)***

**Answer:**

>>CREATE DATABASE IF NOT EXISTS students;

>>USE students;

>>CREATE TABLE student\_info (

Roll\_no INT Primary Key,

first\_name VARCHAR(50) ,

last\_name VARCHAR(50),

dob DATE NOT NULL,

gender CHAR(1) NOT NULL CHECK (gender IN ('M', 'F'))

);

>>CREATE TABLE course (

course\_id INT,

course\_name VARCHAR(100),

credits INT

);

**Q2. Add Constraints using ALTER**

**Using ALTER TABLE, modify the courses table to add:**

**Primary key on course\_id**

**NOT NULL on course\_name**

**Check constraint on credits (must be between 1 and 6)**

**Answer:**

**-- Change couse\_id to primary key**

**ALTER TABLE course MODIFY course\_id INT PRIMARY KEY;**

**-- Change couse\_name to VARCHAR(100) NOT NULL**

**ALTER TABLE course MODIFY course\_name VARCHAR(100) NOT NULL;**

**-- Check constraint on credits (must be between 1 and 6)**

**ALTER TABLE course ADD CONSTRAINT credits\_num CHECK (credits BETWEEN 1 AND 6);**

**Q3. Create a Relationship with Foreign Key**

**Create an enrollments table with:**

**enroll\_id (INT Primary Key)**

**student\_id (INT)**

**course\_id (INT)**

**Add foreign key constraints so that:**

**student\_id references students(student\_id)**

**course\_id references courses(course\_id)**

**Try creating this with and without ON DELETE CASCADE, and note the difference.**

**Q4. ALTER Commands Practice**

**Perform the following changes:**

**Add a new column email (VARCHAR(100)) to the students table.**

**Rename the column dob in students to date\_of\_birth.**

**Drop the column credits from the courses table.**

**Answer:**

**-- Add a new column email (VARCHAR(100)) to the students table.**

**ALTER TABLE student\_info ADD email VARCHAR(100);**

**-- Rename the column dob in students to date\_of\_birth.**

**ALTER TABLE student\_info CHANGE dob date\_of\_birth VARCHAR(50);**

**-- Drop the column credits from the courses table.**

**ALTER TABLE course DROP credits;**

**select \* from course;**

**Q5. DROP vs TRUNCATE**

**Insert a few rows into students and courses.**

**Use TRUNCATE on the enrollments table and observe the difference vs DELETE.**

**Finally, use DROP TABLE to remove the courses table completely.**

**Answer:**

**-- Insert a few rows into students and courses.**

**INSERT INTO student\_info (Roll\_no ,first\_name ,last\_name,date\_of\_birth,gender)**

**VALUES (1,'John', 'Doe', '2022-08-01', 'M'),**

**(2, 'Anita', 'Shah', '2021-06-01', 'F'),**

**(5,'AAA','BBB','2023-09-23','M');**

**INSERT INTO course (course\_id,course\_name)**

**values (102,'Maths'),**

**(104,'BIO');**

**-- Use TRUNCATE on the enrollments table and observe the difference vs DELETE.**

**-- Finally, use DROP TABLE to remove the courses table completely.**

**DROP TABLE course;**